FEATURE-PRESERVING MULTILEVEL HALFTONING BASED ON THRESHOLD DECOMPOSITION

WONG LAI YAN

M.Phil
The Hong Kong
Polytechnic University
2013

The Hong Kong Polytechnic University
Department of Electronic and Information Engineering

Feature-preserving Multilevel Halftoning based on Threshold Decomposition

Wong Lai Yan

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Philosophy

May 2013

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgment has been made in the text.

___________________ (Signed)

Wong Lai Yan (Name of student)

Abstract

Digital halftoning is a technique that converts a continuous image into a bilevel image and is widely used in printing applications. Nowadays, printing technology has advanced a lot and printers are able to print images with ink dots of different intensity levels. Accordingly, a technique for converting a continuous image to a multi-level image is needed. Such a process is generally referred to as multilevel halftoning or multitoning.

Theoretically, multilevel halftoning can be implemented with conventional binary halftoning techniques after replacing their bi-level quantizers with a multilevel quantizer. However, this straightforward extension does not work properly and generally introduces banding artifacts in their outputs. Threshold Decomposition is a widely used technique to remove banding artifacts. It decomposes a continuous image into energy layers, converts them to binary halftones with a conventional binary halftoning algorithm one by one under a stacking constraint, and combines the resultant binary halftones to form a multilevel output. Though this approach provides flexibility and is able to remove banding artifacts effectively, the operation flow introduces a bias to favor the black dots and makes it difficult to preserve both the dark and bright spatial image features in the resultant multilevel halftones.

This work addresses this bias issue and proposes several solutions to remove the bias. Two of them are dedicated for producing three-level multilevel halftones in which the input continuous image is decomposed into two energy layers. In the first solution we process both energy layers simultaneously in an interleaving manner. In the second solution we combine the two energy layers to form a complex energy plane for being processed such that we can take both layers into account at any time of processing. At the end, the idea of the first solution is generalized to take care of more than two energy layers without introducing the bias such that one can produce multitones of any number of output levels. Simulation results show that the proposed solutions can eliminate the bias effectively and produce multitoning outputs of better quality as compared with the conventional multitoning algorithms by preserving the spatial image features faithfully.

Publications arising from the thesis

International Journal Papers

1. L.Y. Wong and Yuk-Hee Chan, “A feature preserving multilevel halftoning algorithm,” *Journal of Electronic Imaging*, 21(4), 043016 (Nov 27, 2012), doi:10.1117/1.JEI.21.4.043016.

International Conference Papers

1. L.Y. Wong and Y.H. Chan, “Preserving Features in Multilevel Halftones,” Proceedings of the Fourth APSIPA Annual Summit and Conference, DEC 3-6, 2012, California, The United State.
2. L.Y. Wong and Y.H. Chan, “Resolving the biases in threshold decomposition-based multitoning,” Proceedings of the 21st European Signal Processing Conference, 9-13 Sep 2012, Marrakech, Marocco.

Acknowledgements

First of all, I would like to express my great appreciation to my supervisor, Dr. Chris Y.H. Chan, for his helpful guidance and patience, professional advice and continuous support over the past two years. He spent his valuable time with me to discuss my research progress and encourage me to make improvement of my study. Without his help, understanding and patience, it is impossible for me to complete this research study.

I would also like to thank my colleague, Dr. Y.H. Fung, who is also supervised by Dr. Chris Y.H. Chan. He shares his research experience, skills and updated information with me, which greatly helps me. Besides, I would also like to thank the people working in Centre for Multimedia Signal Processing. They are very helpful and nice to me.

Furthermore, I would like to express my gratitude to the Department of Electronic and Information Engineering and Centre for Multimedia Signal Processing for providing me a great research environment.

Last but not least, I am deeply grateful to my family, my friends and Mr. K. W. Lau for their love, support and understanding. Without their encouragement and patience, it is impossible for me to complete this research study.

Table of Contents

List of figures

List of tables

List of abbreviations

DBS	Direct Binary Search
FMED	Feature-preserving Multiscale Error Diffusion
g-TD-FMEDi	Generalized interleaved TD-based Feature-preserving Multiscale Error Diffusion
MED	Multiscale Error Diffusion
MSSIM	Mean Structural Similarity Index
SED	Standard Error Diffusion
TD	Threshold Decomposition
TD-CMED	TD based Complex-plane Feature-preserving Multiscale Error Diffusion
TD-FMEDi	Interleaved TD based Feature-preserving Multiscale Error Diffusion
UQI	Universal Objective Image Quality Index

List of symbols

(p,q)	The coordinates of the pixel located to put a dot in a particular iteration when TD-FMEDi, TD-CMED or g-TD-FMEDi is used to produce a multitone
(x,y)	Coordinates of a pixel in format (row,column)
$I(x,y)$	Intensity value of the $(x,y)^{th}$ pixel of image $\boldsymbol{I}$
$\boldsymbol{A}$	Input image for multilevel halftoning
$\boldsymbol{A_d}$	Image of the d^{th} layer after threshold decomposition of $\boldsymbol{A}$
$\boldsymbol{B}$	Output image for multilevel halftoning
$\boldsymbol{B_d}$	Binary halftoning result for layer $\boldsymbol{A_d}$
Budget0	The black dot (0) budget for a binary halftone
Budget1	The white dot (1) budget for a binary halftone
Budget Ratio	The ratio of *Budget0* to *Budget1*
D_b	Black dot budget for multitone $\boldsymbol{B}$
D_w	White dot budget for multitone $\boldsymbol{B}$

$\boldsymbol{E}$	The active energy plane used to search a particular pixel to put a dot in a particular iteration when TD-FMEDi, TD-CMED or g-TD-FMEDi is used to produce a multitone
m	Number of output intensity levels in a multitone
$\boldsymbol{M}$	A Mask used in TD-CMED whose elements report which pixels in the multitoning output have been assigned white or black dots ((True if $M(u,v)=0$))
N	$N=2^a$, the width and the height of a square input image
N_0	Total number of pixels in an image of size $N \times N$
$\boldsymbol{R}$	A Mask used in TD-FMEDi and g-TD-FMEDi whose elements report which pixels in the halftoning output of $\boldsymbol{B_d}$ have been assigned dots (True if $R(u,v)=0$)
S	A parameter used for normalizing a diffusion filter
$w_{s,t}$	The $(s,t)^{\text{th}}$ coefficient of a diffusion filter
Ω	Support of a diffusion filter

Chapter 1

Introduction

1.1 Backgrounds

Digital halftoning is originally a process used to generate a bilevel image to render a continuous-tone image. It is developed for printing images or displaying images with a binary electronic display [Mits92, Ulic87]. In the past three decades, many binary halftoning algorithms were proposed based on various techniques such as thresholding [Maku95], error diffusion [Floy76, Zhou11], neural networks [Crou93], direct binary search [Baqa98], multiscale error diffusion [Chan99, Kats97] and others. Most of them can produce halftones of good quality and are proved to be successful tools to generate binary halftones.

As technology advances, electronic devices are generally able to display images of multiple levels and hence the application of binary halftoning is now mainly on printing. However, the capability of printing devices has also been significantly enhanced recently such that they can now produce outputs of more than two intensity levels [Lin01]. Multilevel halftoning technique is hence required to convert an input image to a multilevel output image for being printed. In general, multilevel halftoning is referred to as multitoning in the field.

In theory, one can easily extend conventional binary halftoning algorithms by replacing the binary quantizer with a multilevel quantizer or by adjusting the screening thresholds to provide multilevel halftones [Gent90, Kang99]. However, when the number of quantization levels is limited, undesirable banding artifacts can generally be found in their outputs as the input levels around an intermediate quantization output level are all mapped to the same output level. In such a case, the gradation reproducibility in the region cannot be maintained.

Various techniques were proposed to remove the banding artifacts. One of the common approaches is to preprocess the image regions that have the potential problem of banding artifacts before multilevel halftoning [Sara12, Yu99]. Another popular approach used in screening algorithms is to control the proportion of adjacent output pixels for the input levels close to an intermediate output level. This can be achieved by developing dedicated multitoning screens according to the desired proportions [Lin02] or adjusting a conventional screen with an improved threshold scaling function [Park07].

Threshold Decomposition is a widely used technique used in multilevel halftoning [Suet08, Rodr08, Fung10]. This technique separates an input image nonlinearly into several energy planes, halftones each one of them with a binary halftoning algorithm, and then combines the binary halftoning results to get a multilevel output. Its application is not limited to screening algorithms. When working with error diffusion-oriented algorithms, Threshold Decomposition generally maintains the gradation reproducibility more effectively than preprocessing and is able to provide outputs of better quality.

Though conventional multitoning algorithms [Suet08, Rodr08, Fung10] based on Threshold Decomposition work effectively to produce multilevel halftoning outputs of good quality, their layer-by-layer nature of operation makes them biased to favor either the brightest or the darkest spatial features of the input images and hence fail to preserve the spatial features well in their halftoning outputs. In this work, we devote our effort to address this issue. Solutions are provided in this thesis such that one can use them to produce multilevel halftoning outputs of even better quality.

1.2 Organization of thesis

This thesis contains six chapters. Chapter 1 provides a brief introduction to multitoning and the issue addressed in the thesis. In particular, we would like to remove the bias introduced by the conventional threshold decomposition technique, which is generally exploited in the multitoning algorithms to remove banding artifacts, to improve the output quality of multitoning.

Chapter 2 provides some background information about halftoning and multitoning. Some classical approaches for their realization are briefly introduced. We put specific emphasis on two techniques called Multiscale Error Diffusion (MED) [Chan98] and Threshold Decomposition (TD) [Suet08]. The former technique allows one to put ink dots on the right positions to preserve the spatial features of the original images, while the latter technique allows one to turn a multitoning process into a sequence of binary halftoning processes to eliminate the banding artifacts commonly found in multitones [Suet08]. The problems of threshold decomposition are analyzed such that one can develop better multitoning algorithms by combining Multiscale Error Diffusion and improved Threshold Decomposition techniques.

For three-level multitoning, conventional TD-based algorithms separate the energy plane of an input image into two layers and then process them sequentially. The processing order introduces bias and affects the quality of the final three-level output. Chapters 3 and 4 introduce two remedial solutions to solve this problem. Specifically, Chapter 3 presents an algorithm which handles the two layers in an interleaving manner to remove the bias. The algorithm proposed in Chapter 4 combines the two layers to form a complex plane for processing such that they can be taken care of at the same time during multitoning.

Chapter 5 extends the idea of the remedial solution presented in Chapter 3 to provide a generalized TD-based multitoning algorithm for producing outputs of any number of levels. It pairs up matched layers and handles the pairs one by one. As layers are carefully matched and the matched layers are processed at the same time in an interleaving manner, the bias can be removed effectively.

Finally, the conclusion of this thesis is given in Chapter 6. We summarize our contributions reported in this thesis and suggest some possible developments of our work in the future.

Chapter 2

Background of multitoning

2.1 Introduction

This chapter provides a brief introduction to multitoning. In Section 2.2, we start our discussion with conventional binary digital halftoning and various realization techniques. We pay particular attention into error diffusion techniques such as MED [Chan98] as they are effective in providing good binary halftoning outputs. Our discussion is then extended to multitoning and its realization in Section 2.3. Focus is put on threshold decomposition, a technique that is widely used in many multitoning algorithms to remove banding artifacts. Its implementation and its potential problems are discussed in details. Finally, a brief summary of this Chapter is provided in Section 2.4.

2.2 Digital halftoning

Digital halftoning is a process that uses bilevel pixels to simulate a gray-scale image on a bilevel output device. It is a necessary process in printing applications. There are many algorithms for realizing digital halftoning [Dame01, Hwan04, Lau98, Lau03, Li00, Pang08]. They can be roughly classified into one of the following three categories: screening [Maku95, Mits91, Schr86, Ulic93, Zeng00], error diffusion [Floy75, Jarv76, Lin97, Stev85, Stuc81] and iterative process [Guo11, Papp03, Zhen10].

Screening is basically a point process. For each pixel, it compares its intensity value against a threshold value to determine its output level. The threshold value can be spatially invariant [Klen70] or variant adaptive [Leon92]. It can also be adaptive [Baye73] or non-adaptive [Gran90] to the local region of a pixel. Screening is very time efficient, but the output image quality is generally low.

Algorithms based on error diffusion process the image pixels one by one according to a specific order [Floy76]. For each pixel being processed, its intensity value is quantized to either 0 or 1. The quantization error of the pixel is then diffused to the pixel's selected neighbors. Accordingly, the intensity values of the selected neighboring pixels are adjusted such that the error diffused into them can be taken into account when they are processed in the future. The processing order of the pixels can be fixed [Floy75] or input dependent [Chan99, Kats97]. As compared with screening algorithms, error diffusion-based algorithms can provide outputs of better quality, but they are more time consuming.

The remaining algorithms are basically iterative algorithms. Starting with a preliminary estimate of the binary halftoning output, these algorithms iteratively adjust the binary output to minimize a predefined quality measure derived based on the human visual system. The realization can be based on neural networks [Anas88, Ostr00], simulated annealing [Geis93], and Direct Binary Search (DBS) [Papp03]. It takes time for the output to converge and hence their computational complexity is huge. However, they generally provide the best halftoning output in terms of their corresponding adopted quality measure due to their operation nature.

Among the various techniques used in halftoning, Feature-preserving Multiscale Error Diffusion (FMED) is proved to be one of the best algorithms in preserving the spatial features of an input image and producing the halftoning outputs of desirable noise characteristics [Chan04]. As the issue to be addressed in this project is on how to produce high-quality multitones, FMED is naturally selected to be one of the key components when developing the multitoning algorithms proposed in this thesis. FMED [Chan04] stems from Standard Error Diffusion (SED) [Floy76] and MED [Kats97]. In the following part of this section, we will have a more detailed revision on Standard Error Diffusion [Floy75], MED [Kats97] and FMED [Chan04].

2.2.1 Standard Error Diffusion

Standard Error Diffusion (SED) was introduced in 1976 by Floyd and Steinberg [Floy76]. It is able to provide a halftoning output of better quality as compared with screening methods at a cost of slight increase of computational effort.

Figure 2.1 shows the operation flow of SED. In particular, it raster-scans an input image and processes the scanned pixels one by one until the whole image is processed. For each pixel it scans, its intensity value is quantized and the quantization error is diffused to its not-yet-processed neighboring pixels with a causal diffusion filter. The error diffused to a pixel changes the pixel's intensity value such that the quantization carried out for the pixel in the future can take the error into account. The diffusion filter determines which neighboring pixels are affected and how they are affected. Figure 2.2 shows a diffusion filter proposed by Floyd and Steinberg [Floy76].

As the pixels are processed according to a predefined order and a causal diffusion filter is used in SED, there is a bias in the direction of error diffusion and hence a directional hysteresis is resulted in the output. Various remedial solutions were proposed to remove this problem by modifying the scanning paths and diffusion filters [Lau08, Mese02]. However, due to the nature of their working principle, a fixed predefined scanning order and a causal filter must be used and hence the directional hysteresis cannot be completely eliminated from the root.

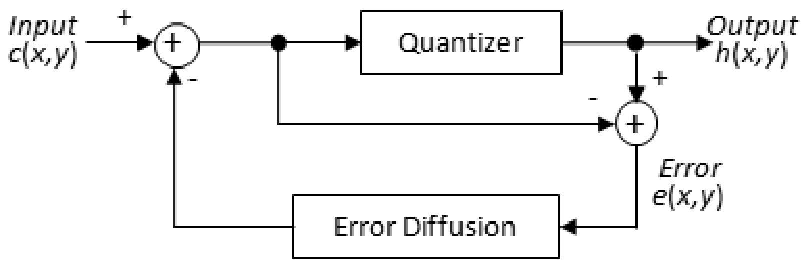


Figure 2.1 Operation flow of Standard Error Diffusion

	X	7/16
3/16	5/16	1/16

Figure 2.2 Causal diffusion filter used in Standard Error Diffusion

2.2.2 Multiscale Error Diffusion

In 1997, Katsavounidis et al. proposed a technique called Multiscale Error Diffusion (MED) [Kats97] to remove the directional hysteresis. To a certain extent, MED combines error diffusion and iteration method to provide a better halftoning result than the conventional error diffusion methods at an acceptable computational cost. Unlike the conventional iteration processes such as Direct Binary Search (DBS) [Papp03], the number of required iterations is bounded by the total energy of the input image.

MED is a two-step iterative method. At the very beginning, an energy plane and an output plane are created and they are, respectively, initialized to be the input image and a black plane of equal size. In each iteration, it searches for the currently best location in the output plane to assign a white dot. The search is based on the rule of “maximum intensity guidance”. In particular, starting with the whole energy plane as the region of interest, it repeatedly divides the region of interest into four sub-regions of equal size and selects the one with the maximum sum of intensity values as the new region of interest until a single pixel is reached. The selected pixel is then quantized to 1 and the corresponding pixel location in the output plane is assigned a white dot. The energy plane is then updated by diffusing the quantization error of the selected pixel to the pixel's neighbors with a non-causal diffusion filter. Figure 2.3 shows a typical non-causal diffusion filter used in MED as an example.

By repeating the above procedures, white dots are assigned to the output plane one by one until the white dot budget is used up. The white dot budget is defined as the nearest integer of the total sum of the intensity values of the input image. It determines the total number of white dots to appear in the output image. This explains why the number of iterations is bounded by the total energy of the input image.

As there is no predefined scanning order to follow when MED processes the pixels and quantization errors are diffused in all directions with a non-causal filter, the directional hysteresis can be eliminated when MED is used to produce binary halftones.

$$\begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & X & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array} \times 1/12$$

Figure 2.3 Non-causal diffusion filter used in Katsavounidis et al.'s MED

2.2.3 Feature-preserving Multiscale Error Diffusion

In the original version of MED [Kats97], the quantization error of a processed pixel is diffused in all directions. It is possible that some of the neighbors of the quantized pixel have already been processed, so the error diffused into them will be trapped forever and eventually degrades the quality of the halftoning output. To solve this problem, the quantization error should only be diffused into the neighbors that have not been processed.

In general, the pixels handled earlier have more freedom to assign dots of appropriate output levels. The critical dots which highlight the local features of the original input image should hence processed first. Chan et al. found that the minority dots in a local region of a halftone were critical to show the local features and modified MED accordingly to preserve image features [Chan04]. This was referred to as Feature-preserving Multiscale Error Diffusion (FMED) in their work [Chan04].

Like MED, FMED is also a two-step iterative algorithm. In each iteration, it searches for the currently most critical location in the output plane to put a dot. The searching is based on “extreme error intensity guidance” instead of the “maximum error intensity guidance”. In particular, starting with the whole energy plane as the region of interest, it repeatedly divide the region of interest into nine overlapped sub-regions of equal size and selects the one with the maximum sum of intensity values as the new region of interest until the region of interest is small enough. At this point, if the average energy of the region of interest is above 0.5, it will adjust the selection criterion to pick the sub-region with the minimum sum of intensity values. Then it continues the divide-and-select step to update the region of interest until a single pixel is reached. According to the quantization result of the selected pixel in the energy plane, an appropriate dot is assigned to the corresponding pixel of the output plane. The energy plane is then updated by diffusing the quantization error of the selected pixel to the pixel’s not-yet-processed neighbors. Since there is no predefined processing order of pixels, statistically there is no bias on the diffusion direction even though some quantization errors may not be diffused in all directions.

The rule of “extreme error intensity guidance” allows one to search for the pixels that highlight the local features in an input image and then put the appropriate dots at the corresponding locations in the output plane. These dots are put onto the output plane first and hence the critical dots are easier to be put at the right positions to represent the image details correctly. These dots are generally minority dots in a local region.

By putting the right dots at the right positions, FMED effectively preserves the spatial features of an image and is able to provide a better output in terms of various quality measures as compared with any other error diffusion-based halftoning algorithms [Floy76, Kats97].

2.3 Digital multitoning

Printer technology has advanced a lot in recent years and current printers are able to produce ink dots of more than one intensity levels. Obviously, rendering an input image with more gray levels can reduce the step size between gray levels in the output image and hence the quality of the resultant image can be closer to that of the original input image. Multitoning technique is an advanced halftoning method used to render an input image with more output levels. In this section, we will address an artifact that can be easily found in the multitones produced with the conventional digital multitoning algorithms and review some solutions proposed to eliminate the artifact.

2.3.1 Banding artifact

In theory, one can easily extend the conventional binary halftoning algorithms to produce multilevel halftones by replacing the binary quantizer with a multilevel quantizer or by adjusting the screening thresholds to provide multilevel halftones [Gent90, Kang99]. However, when the number of quantization levels is limited, undesirable banding artifacts can generally be found in their outputs as the input levels around an intermediate quantization output level are all mapped to the same output level [Rodr08]. In such a case, the gradation reproducibility in the region cannot be maintained as shown in Figures 2.4 and 2.5.

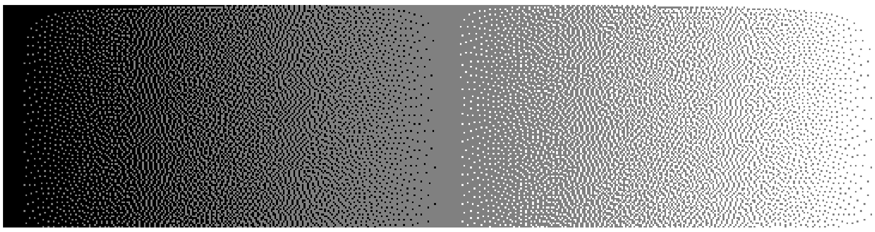


Figure 2.4 Banding artifacts in three-level multitoning output of a ramp image

Figure 2.5 Three-level multitone bearing banding artifacts

Various techniques have been proposed to remove the banding artifacts. One of the common approaches is to preprocess the image regions that have the potential problem of banding artifacts before multilevel halftoning [Sara12, Yu99]. Another popular approach used in the screening algorithms is to control the proportion of the adjacent output pixels for the input levels close to an intermediate output level. This can be achieved by developing the dedicated multitoning screens according to the desired proportions [Lin02] or adjusting a conventional screen with an improved threshold scaling function [Park07]. In next sub-section, we will have a brief review on Threshold Decomposition, which is also a technique widely used to eliminate the banding effect.

2.3.2 Threshold Decomposition

Threshold Decomposition is a widely used technique for removing the banding artifacts in multilevel halftoning [Suet08, Rodr08, Fung10]. This technique separates an input image nonlinearly into several energy planes, halftones each one of them with a binary halftoning algorithm, and then combines the binary halftoning results to obtain a multilevel output as shown in Figure 2.6. As there is no limitation on the selection of the binary halftoning algorithm used to halftone an energy plane, the application of threshold decomposition is not limited to the screening algorithms. When working with the error diffusion-oriented algorithms, threshold decomposition generally maintains the gradation reproducibility more effectively than the preprocessing approaches mentioned in Section 2.3.1 and is able to provide the outputs with better quality.

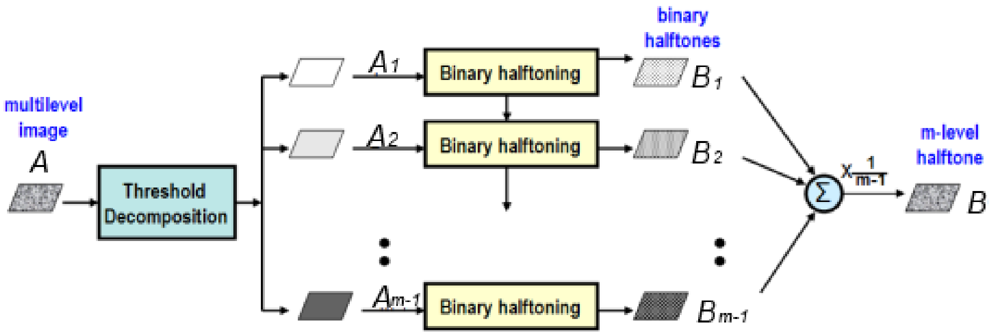


Figure 2.6 Operation flow of Threshold Decomposition-based multilevel halftoning

Consider the case that we want to convert a gray-level image $\boldsymbol{A}$ into a multilevel image $\boldsymbol{B}$. The pixel values of $\boldsymbol{A}$ are bounded in [0,1], where 0 and 1 denote the minimum (black) and maximum (white) intensity levels, respectively. In contrast, the pixel values of $\boldsymbol{B}$ are confined to be a member of $\{r/(m\text{-}1) \mid r=0, \ldots, m\text{-}1\}$, where m is the number of available output intensity levels. Without loss of generality, we assume that the size of images $\boldsymbol{A}$ and $\boldsymbol{B}$ is $N \times N$, where $N=2^a$ and a is a positive integer. Note that the size of the input image is not limited to $N \times N$ when the algorithm is applied. It is used for illustration only. For reference, $I(x,y)$ denotes the pixel value of image $\boldsymbol{I}$ at position (x,y).

When threshold decomposition is applied, the input image $\boldsymbol{A}$ is first decomposed into $m-1$ images, each of which is denoted as $\boldsymbol{A_d}$ for $d = 1, \ldots, m-1$, such that we have

$$A(x,y) = \sum_{d=1}^{m-1} A_d(x,y)/(m-1) \qquad \text{for } x,y = 0,1,2,\ldots,2^a\text{-}1 \qquad (2.1)$$

Theoretically, one can adopt any decomposition scheme to determine $\boldsymbol{A_d}$ for d = 1,2, … ,m-1. A widely used decomposition scheme was suggested by Suetake et al. [Suet08] to determine $\boldsymbol{A_d}$ as

$$A_d(x,y) = A_{d-1}(x,y) - A(x,y)^{d-1} \times \left(\frac{(m-1)!}{(d-1)!(m-d)!}\right)(1 - A(x,y))^{m-d}$$

for d = 1,2, … ,m-1 (2.2)

where $A_0(x,y)$ = 1 for all (x,y).

With Eqn. (2.2), we can decompose an input image into m – 1 images (also referred to as layers) with different intensity scales. A layer of smaller index value carries more energy and is referred to as a higher energy layer while a layer of larger index value is referred to as a lower energy layer for reference purpose hereafter. Figure 2.7 plots the decomposition curves defined by Eqn. (2.2) for three-level and five-level multitoning as examples.

After threshold decomposition, the decomposed layers are processed from the layer of the highest energy to the layer of the lowest energy one by one subject to a stacking constraint such that the ink dots can be distributed homogenously in the intermediate-tone regions of an image to eliminate the banding artifacts [Suet08]. In particular, when a location in a higher energy layer is assigned a black pixel in its binary output, black pixels must also be assigned to the corresponding locations in the binary outputs of the lower energy layers. Figure 2.8 shows an example of how a lower energy layer is affected by a higher energy layer under the stacking constraint when producing a four-level multitone.

This stacking constraint confines the dot assignment in lower energy layers. The lower the energy of a layer, the more restriction it has to assign a white dot. As the final multilevel output is the sum of the binary halftoning results of all layers, the locations of the brighter dots in the final multilevel result are relativity more restricted than the locations of the darker dots. Hence, it is more difficult to put the bright dots at the right positions to preserve the bright features in an image. In other words, the darker dots have more freedom than the brighter dots to be assigned to appropriate locations to show the corresponding features. This results in a bias that it does not favor the preservation of bright features.

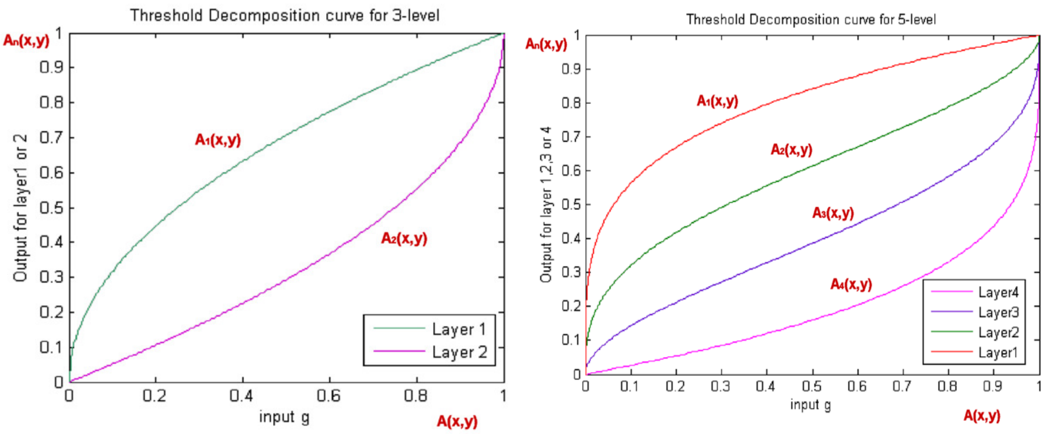


Figure 2.7 Threshold decomposition curves suggested in [Suet08] for (a) three-level and (b) five-level multitoning

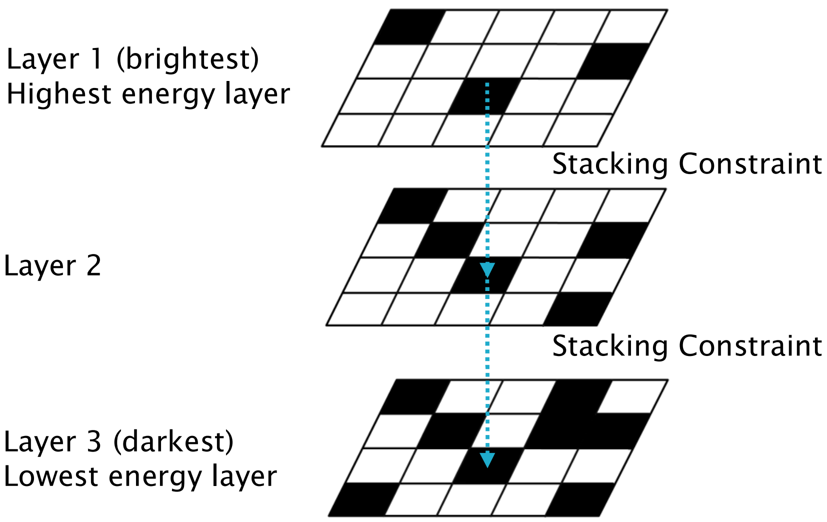


Figure 2.8 Example of stacking constraint in four-level multitoning

2.4 Summary

This chapter reviews some techniques used in binary halftoning. Focus is put on SED, MED and FMED. Among the various binary halftoning algorithms, FMED is able to preserve the spatial features and provide desirable noise spectral characteristics simultaneously, and so it is selected as a tool to develop the feature-preserving multitoning algorithms in this thesis.

Straightforward extensions of conventional binary halftoning algorithms cannot work properly to produce the multitones of high quality. Banding artifacts are commonly found in their outputs. Threshold Decomposition is a technique which can provide flexibility and eliminate the artifacts at the same time. However, it introduces bias to the darker dots due to its nature of operation flow and hence the bright and dark spatial features in an image cannot be equally preserved. In the subsequent chapters, some remedial solutions are provided and the corresponding multitoning algorithms are proposed accordingly.

Chapter 3

Three-level TD-based multitoning by processing layers in an interleaving manner

3.1 Introduction

As mentioned in Section 2.3, when Threshold Decomposition (TD) is applied to do multitoning, the energy layers are processed one by one using a binary halftoning algorithm from the highest energy layer to the lowest energy layer subject to a stacking constraint. By doing so, the positions of the black dots in the final multilevel halftone are determined first in the conventional approach. This bias makes the conventional TD-based algorithms difficult to preserve the bright features in the image and hence lowers the output quality.

In order to remove the bias introduced by the conventional TD-based algorithms, we suggest handling the layers with maximum energy and minimum energy simultaneously to locate the brightest and the darkest dots in the final multilevel halftone in an interleaving manner. To simplify the situation, in our preliminary investigation of the issue, a feasibility study is carried out in which the number of output levels of a multitone is limited to be three.

The organization of this chapter is as follows. After a brief introduction, Section 3.2 presents a three-level multitoning algorithm which is developed based on the aforementioned idea. The performance of the proposed algorithm is evaluated using computer simulations and the results are provided in Section 3.3. A summary of this chapter is finally given in Section 3.4.

3.2 TD-based multitoning with interleaving

Figure 3.1 shows a framework of conventional Three-level TD-based multitoning algorithm. When the number of output levels of a multitone is limited to 3, Eqns. (2.1) and (2.2) can be rewritten as

$$A(x,y)=\frac{A_1(x,y)+A_2(x,y)}{2} \qquad \text{for } x,y=0,1,\ldots,2^a\text{-}1 \qquad (3.1)$$

where $A_1(x,y)=2A(x,y)-A(x,y)^2$ (3.2)

and $A_2(x,y)=A(x,y)^2$ for all (x,y) (3.3)

Note that layer $\boldsymbol{A_1}$ carries more energy and layer $\boldsymbol{A_2}$ carries less energy after decomposition. In order words, layer $\boldsymbol{A_1}$ is brighter than layer $\boldsymbol{A_2}$.

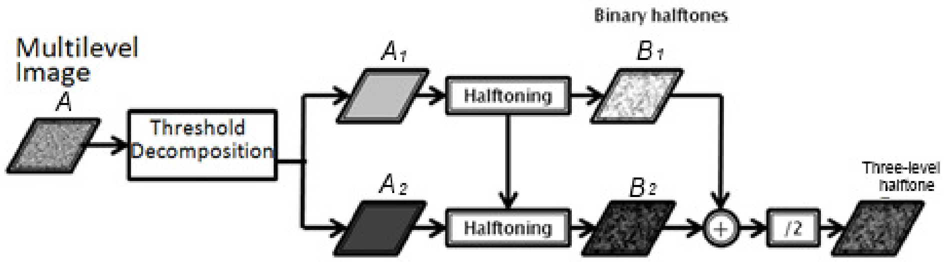


Figure 3.1. Operation flow of conventional three-level TD-based multitoning

As layer $\boldsymbol{A_1}$ is processed first, the black dots in the final multilevel halftone are determined first in the conventional approach and the bright feature in the image is more difficult to preserve in the resultant multitone. In order to remove this bias, we suggest handling the two energy layers simultaneously to locate the brightest and the darkest dots in the final multilevel halftone in an interleaving manner.

In theory, one can apply any binary halftoning algorithm to halftone a layer after threshold decomposition. In our proposed algorithm, FMED [Chan04] and its modified version are used to halftone the layers. Since FMED is more flexible to introduce dots of different nature to the outputs of individual layers, it is superior to other conventional binary halftoning algorithms in preserving the spatial features.

As mentioned in Chapter 2, FMED is a two-step iterative method developed based on MED [Chan98] which can provide a good quality for the resultant image. We adopt the general idea of FMED and modify it to halftone the layers.

Figure 3.2 shows the operation flow of the proposed algorithm for producing a three-level output. Instead of processing layers $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ one by one sequentially, the proposed algorithm processes the two layers simultaneously to locate the darkest and the brightest pixels in the final multilevel halftone in an interleaving manner.

At the beginning of the halftoning process, we need to determine the total number of black dots and white dots that should be placed in $\boldsymbol{B}$. They are, respectively, referred to as black dot budget and white dot budget. Specifically, they can be estimated as follows.

$$Budget0 = N_0 - \sum_{(x,y)} A_1(x,y)$$

$$Budget1 = \sum_{(x,y)} A_2(x,y) \tag{3.4}$$

where N_o=N×N is the total number of pixels in the image. Accordingly, a budget ratio is defined as

$$Budget\ Ratio = \frac{\sum_{(x,y)} A_2(x,y)}{N_0 - \sum_{(x,y)} A_1(x,y)} \tag{3.5}$$

Then we alternately locate a pixel in layer A_1 to assign a '0' and locate a pixel in layer A_2 to assign a '1' under the guidance of the budget ratio. Under the stacking constraint, this is equivalent to alternately placing the darkest and the brightest dots in the final multilevel halftoning output. By doing so, the bias in either white or black dots is removed and a balance on preserving the dark and bright spatial features in the image can be achieved.

When the dots are put in the layers, the black dot budget for layer A_1 and the white dot budget for layer A_2 are consumed accordingly. At the time when we determine which type of dots should be placed next, we pick the one which can make the ratio of the remained white dot budget to the remained black dot budget closer to the original budget ratio.

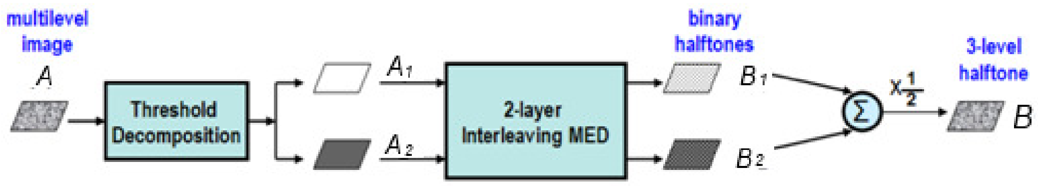


Figure 3.2 Operation flow of TD-FMEDi for producing three-level halftone

The search of the pixel to put a dot is based on an energy plane E. To locate a pixel to put a black dot, the energy plane E is initialized to be layer A_1. Otherwise it is initialized to be layer A_2. The search is carried out as follows. Starting with the selected energy plane E as the region of interest, we repeatedly divide the region of interest into nine overlapped sub-regions of equal size and select one of these sub-regions to be the new region of interest based on a selection criterion. That sub-regions are overlapped is for eliminating blocking artifacts. When we are looking for a pixel in layers $\boldsymbol{A_1}$ to put a black dot, we follow the minimum intensity guidance and pick the sub-region having the smallest sum of its pixel values. Otherwise, we follow the maximum intensity guidance and pick the one of the largest sum. We repeat the above steps to update the region of interest until a pixel location is reached.

Without loss of generality, let $\boldsymbol{A_d}$ be the layer that we select to locate a pixel to put a dot and the selected location of the pixel is (p,q). According to the stacking constraint, when a pixel in the brighter layer ($\boldsymbol{A_1}$) is selected to put a black dot in its binary halftone, a black dot must also be assigned to the corresponding pixel in the binary halftone of the darker layer ($\boldsymbol{A_2}$) to maintain consistency. Based on the same philosophy, when a pixel in the darker layer ($\boldsymbol{A_2}$) is selected to put a white dot in its binary halftone, a white dot must also be assigned to the corresponding pixel in the binary halftone of the brighter layer ($\boldsymbol{A_1}$). In other words, once pixel (p,q) is selected, no matter whether the search is based on layer $\boldsymbol{A_1}$ or $\boldsymbol{A_2}$, both layers $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ are affected and the dots with identical intensity level are put at $\boldsymbol{B_1}(p,q)$ and $\boldsymbol{B_2}(p,q)$ simultaneously as follows.

$$B_1(p,q) = B_2(p,q) = \begin{cases} 1 & if\ searching\ is\ based\ on\ A_2 \\ 0 & if\ searching\ is\ based\ on\ A_1 \end{cases} \tag{3.6}$$

where $\boldsymbol{B_d}$ is the binary output plane of layer $\boldsymbol{A_d}$.

For each layer $\boldsymbol{A_i}$, where $i \in \{1,2\}$, the difference between $B_i(p,q)$ and $A_i(p,q)$ is then diffused to $A_i(p,q)$'s neighbors to update layer $\boldsymbol{A_i}$ as follows.

$$A'_i(x,y) = \begin{cases} 0 & if\ (x,y) = (p,q) \\ A_i(x,y) - w_{x-p,y-q} \cdot \big(B_i(p,q) - A_i(x,y)\big) \cdot \frac{R(x,y)}{S} & else \end{cases} \tag{3.7}$$

where $A'_i(x,y)$ and $A_i(x,y)$ are, respectively, the values of pixel (x,y) in layer i after and before the error diffusion process, $B_i(p,q)$ is the value assigned to pixel (p,q), $R(x,y)$ is a mask defined as

$$R(x,y) = \begin{cases} 0 & if\ B_i(x,y)\ has\ been\ assigned\ a\ dot \\ 1 & else \end{cases} \tag{3.8}$$

$\boldsymbol{w_{s,t}}$ for $(s,t) \in \Omega$ is a filter weight of a non-causal diffusion filter with support **Ω**, and

$$S = \sum_{(x-p,y-q)\in\Omega}\left(w_{x-p,y-q} \cdot R(x,y)\right) \tag{3.9}$$

By diffusing the difference between $B_i(p,q)$ and $A_i(p,q)$ with Eqn.(3.7), the difference can be shared by pixel (p,q)'s neighboring pixels and compensated for later.

The aforementioned procedures are repeated until we use up all black dot budget for the higher energy layer $\boldsymbol{A_1}$ and all white dot budget for the lower energy layer $\boldsymbol{A_2}$. Values of some pixels in $\boldsymbol{B_2}$ and $\boldsymbol{B_1}$ may be left unassigned at this point. They are backfilled as follows to complete the construction of the binary halftones of layers $\boldsymbol{A_2}$ and $\boldsymbol{A_1}$.

$$B_1(x,y) = 1 \qquad \text{for all } (x,y) \text{ in } \boldsymbol{B_1} \text{ not assigned } 0 \tag{3.10}$$

and

$$B_2(x,y) = 0 \qquad \text{for all } (x,y) \text{ in } \boldsymbol{B_2} \text{ not assigned } 1 \tag{3.11}$$

Finally, the multilevel halftone is obtained by combining the two binary halftones as follows.

$$B(x,y) = \frac{1}{2}\left(B_1(x,y) + B_2(x,y)\right) \tag{3.12}$$

Figure 3.3 summarizes the operation flow of the proposed multilevel halftoning algorithm in a form of pseudo code. As the proposed three-level TD-based multitoning algorithm is based on FMED and the layers are processed in an interleaving manner, it is referred to as TD-FMEDi in this thesis.

```
Initialize Budget1 and Budget0 with Eqns. (3.4)
Compute BudgetRatio = Budget1/Budget0;
WHILE Budget1+Budget0>0
  Determine if white or black dot should be put in this round
  IF white dot should be put
      Search for a pixel location in A2 via maximum intensity guidance
      B1(xo,yo) =B2(xo,yo) = 1      % Assume that the location is (xo,yo)
      Diffuse error of Ai(xo,yo) to its neighbors in Ai for i=1,2
      Budget1 = Budget1 - 1
  ELSE
      Search for a pixel location in A1 via minimum intensity guidance
      B1(xo,yo) =B2(xo,yo) = 0      % Assume that the location is (xo,yo)
      Diffuse error of Ai(xo,yo) to its neighbors in Ai for i=1,2
      Budget0 = Budget0 - 1
  END
END
B1(x,y) = 1      for all (x,y) in B1 not assigned 0
B2(x,y) = 0      for all (x,y) in B2 not assigned 1
```
Multilevel halftone is $\mathrm{B}(x, y) = \frac{1}{2}\sum_{i=1,2} \mathrm{B}_i(x,y)$

Figure 3.3 Pseudo code of TD-FMEDi algorithm

3.3 Simulation results and discussion

Simulation was carried out to evaluate the performance of the proposed method with nine standard 256 gray-level testing images [Spri] shown in Appendix including *Lena*, *Boat*, *Goldhill*, *Barbara*, *Airplane*, *Peppers*, *Baboon, Girl* and *Man*. Each one is of size 512×512. For comparison, the results of three additional conventional threshold decomposition-based multilevel halftoning algorithms ([Suet08], [Rodr08] and [Fung10]) are also evaluated in the simulation. In particular, Suetake et al.'s algorithm [Suet08] halftones each layer with standard error diffusion using Floyd Steinberg's diffusion filter with serpentine scanning while Fung et al.'s algorithm [Fung10] halftones each layer with FMED. Both algorithms adopt the decomposition scheme specified in Eqns. (3.2) and (3.3) as the proposed algorithm does. In contrast, Rodríguez et al.'s algorithm [Rodr08] exploits another decomposition scheme to provide desired noise characteristics according to their suggested blue noise multitoning structure. We also include Sarailidis et al.'s algorithm [Sara12] in the comparison as a reference though it is not a threshold decompositon-based multilevel halftoning algorithm. It is included because it also stems from MED [Kats97] as the proposed algorithm does.

Figures 3.4 and 3.5 show some regions of the three-level halftoning results of two testing images for subjective evaluation. The original image shown in Figure 3.4(a) carries a lot of image details. Whether these details can be faithfully reported in the halftoning output is a challenge to the evaluated algorithms. In this aspect, the proposed algorithm performs the best and Fung et al.'s algorithm follows. One can see that they can report the texture of the rooftop of the house in white, the window frames, the white line behind the head of the pedestrian, the vertical drain pipe in the left middle of the image and its shadow on the wall better than the others. From Figure 3.4(c), one can even see the white dot on the right top of the image which marks a cow in the field behind the row of terraced houses. This feature is missing in all other algorithms' outputs.

The image details of the original image shown in Figure 3.5(a) are not as rich as that shown in Figure 3.4(a). However, the proposed algorithm can still provide an output of better quality. This can be verified by inspecting the U.S. air force roundel logos and the flight numbers printed on the aircrafts shown in Figure 3.5. We note that there are a lot of smooth regions in the original image shown in Figure 3.5(a) and they can also be rendered well with the proposed algorithm without being sharpened. Similar observations can be found in the simulation results of other testing images shown in the Appendix.

Figure 3.6 shows the halftoning results of a gray level ramp strip. The output of the proposed algorithm offers a smooth transition and there is no banding artifact found in its output.

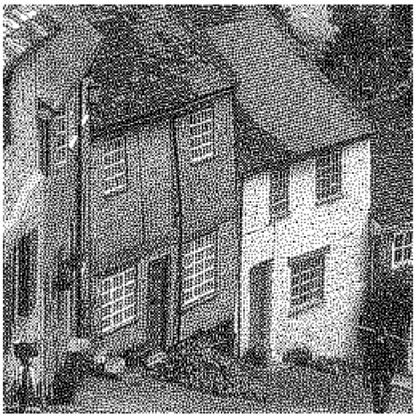
Both Fung et al.'s and the proposed algorithm are FMED-based multilevel halftoning algorithms and adopt the same threshold decomposition scheme. Their only difference is that Fung et al.'s handles the layers one by one while the proposed algorithm handles them in an interleaving manner. This makes the proposed algorithm be able to preserve both the bright and dark features in a way better than Fung et al.'s algorithm. As evidence, Figure 3.7 shows the halftoning results of a combined artificial testing image having only four input gray levels. To a certain extent, both the bright and dark characters in Figure 3.7(b) are more recognizable than those in Figure 3.7(c).

The complexity of FMED is proportional to the number of pixels in an image and most of the effort is for searching locations to put the dots. When FMED works with the conventional TD framework to produce a multilevel halftone, the total searching effort is roughly proportional to the number of layers. This is because it performs FMED once for each layer and the searching effort of FMED is proportional to the number of pixels in the input image. However, when it works with the proposed framework, the total number of pixel locations that we have to locate is bounded by the number of pixels in the image, which is independent of the number of layers. For the production of a Three-level halftone, a reduction ratio of 2 to 1 can be achieved.

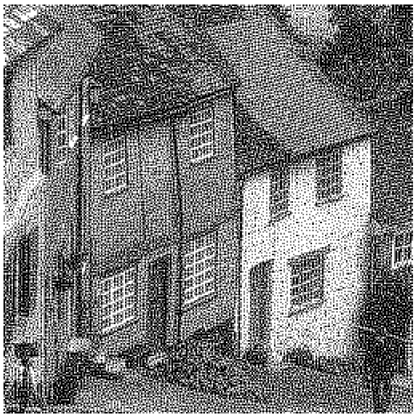
In our Matlab simulation, the running time of TD-FMEDi is more or less the same as that of Fung et al.'s algorithm [Fung10]. As compared with the running time of Rodriguez et al.'s [Rodr08] and that of Suetake et al.'s [Suet08], the ratio is 700:1:1. Since the realization of all algorithms was not optimized in our simulation, the complexity comparison reported here may not reflect the real situation accurately.

(a) Original

(b) Rodríguez et al. [Rodr08]



(c) TD-FMEDi

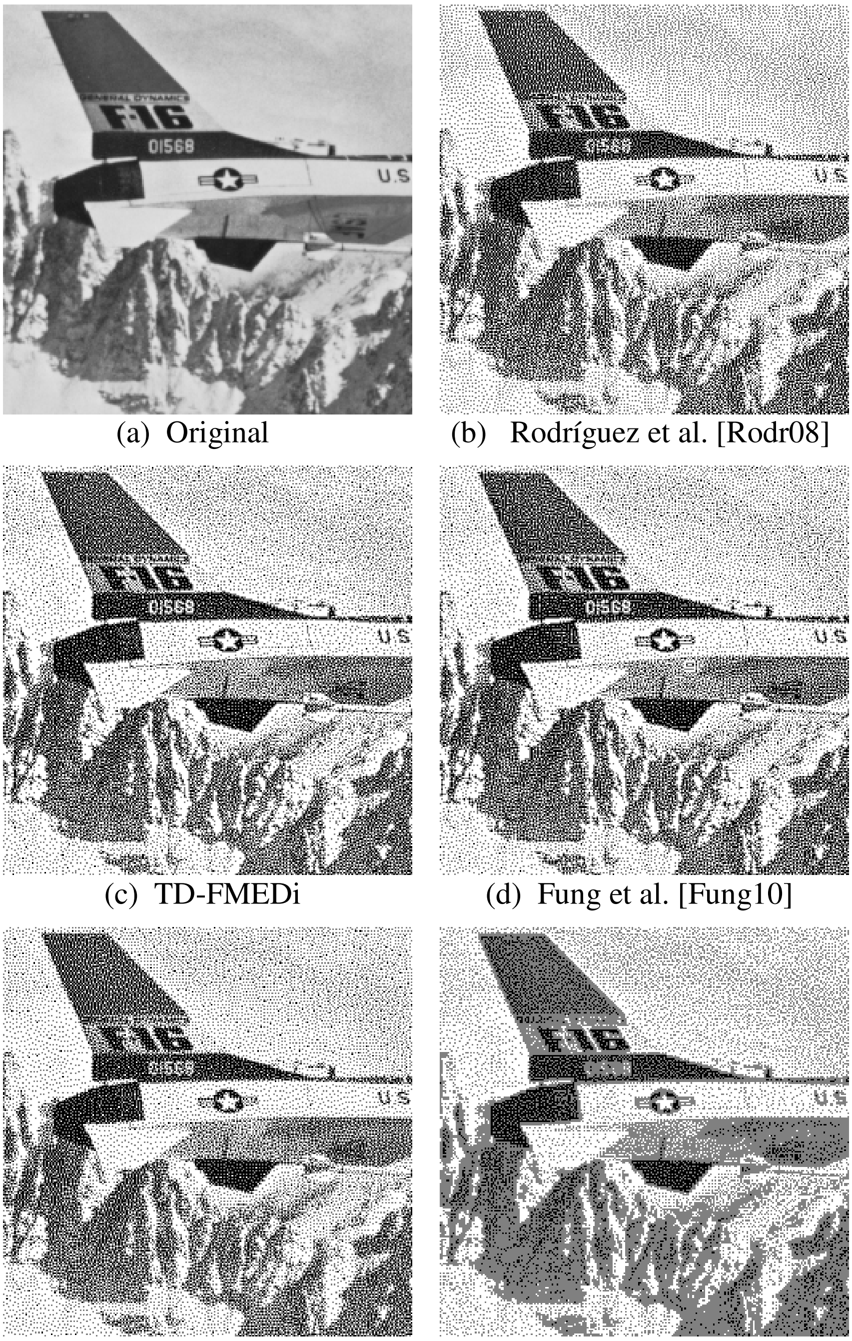


(d) Fung et al. [Fung10]

(e) Suetake et al. [Suet08]

(f) Sarailidis et al. [Sara12]

Figure 3.4 Parts of three-level halftoning results of *Goldhill*: (a) Original, (b) Rodríguez et al.[Rodr08], (c) TD-FMEDi algorithm, (d) Fung et al.[Fung10], (e) Suetake et al.[Suet08] and (f) Sarailidis et al.[Sara12]



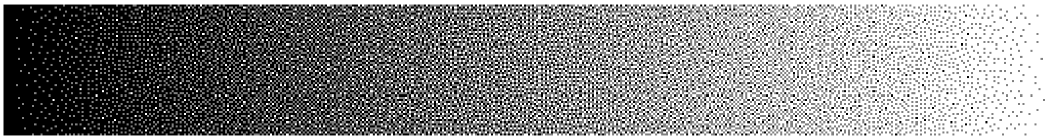
(a) Original (b) Rodríguez et al. [Rodr08]

(c) TD-FMEDi (d) Fung et al. [Fung10]

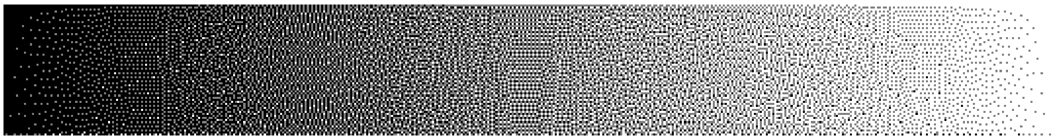
(e) Suetake et al. [Suet08] (f) Sarailidis et al. [Sara12]

Figure 3.5 Parts of three-level halftoning results of *Airplane*: (a) Original, (b) Rodríguez et al.[Rodr08], (c) TD-FMEDi algorithm, (d) Fung et al.[Fung10], (e) Suetake et al.[Suet08] and (f) Sarailidis et al.[Sara12]

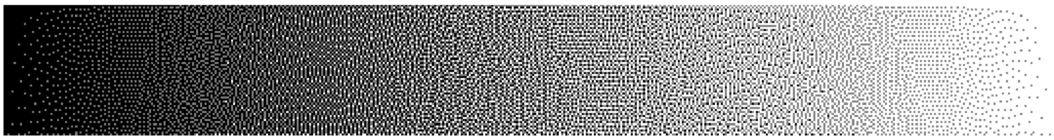
Original



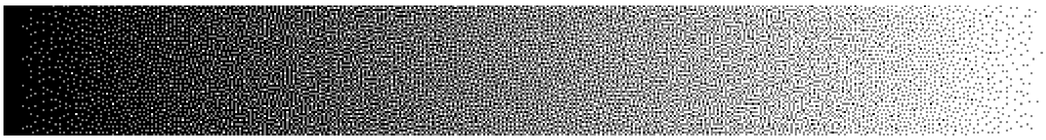
TD-FMEDi



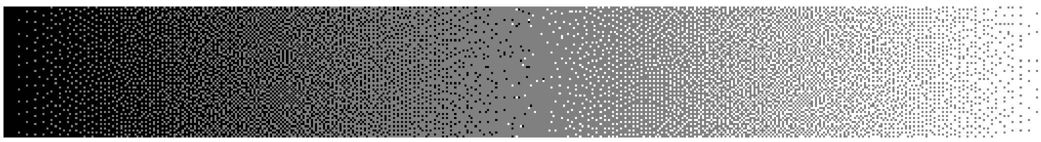
Suetake et al. [Suet08]



Rodríguez et al. [Rodr08]

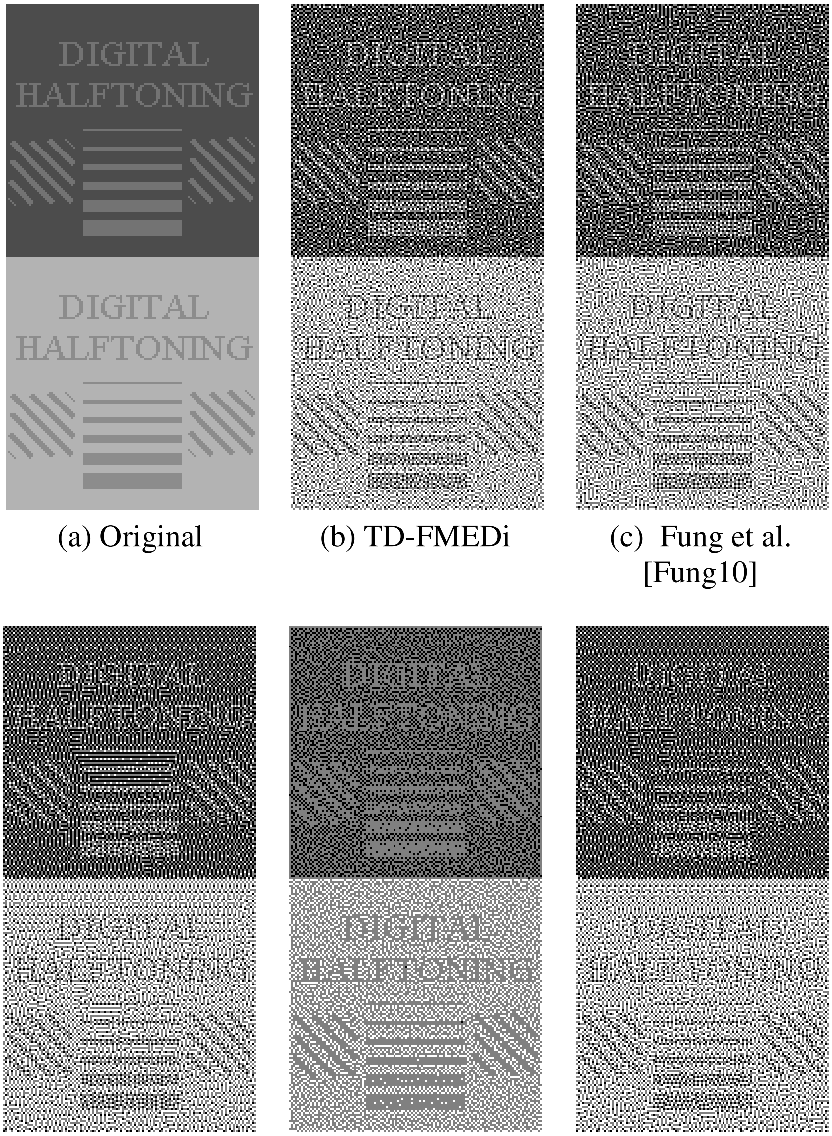


Fung et al. [Fung10]



Sarailidis et al. [Sara12]

Figure 3.6 Three-level halftoning outputs of gray-scale ramp obtained with different algorithms



(a) Original

(b) TD-FMEDi

(c) Fung et al. [Fung10]

(d) Suetake et al. [Suet08]

(e) Sarailidis et al. [Sara12]

(f) Rodríguez et al. [Rodr08]

Figure 3.7 Three-level halftoning outputs of artificial testing images

3.4 Summary

Conventional three-level TD-based multitoning algorithms decompose an input image into two layers and process them one by one. The brighter layer is handled first and hence pixels that are of level 0 in the final three-level multitoning output are positioned first. Bright spatial features are difficult to preserve because of this bias.

In this chapter, we have proposed an approach which processes the two layers simultaneously in an interleaving manner. The bias is then automatically removed and the visual quality of the output is improved. The suggested approach can be easily realized with FMED [Chan04]. Simulation results have showed that the proposed method can provide a better result than the conventional TD-based algorithms such as [Suet08, Fung10] in terms of both subjective and objective criteria.

Chapter 4

Three-level TD-based multitoning by combining layers to process

4.1 Introduction

The stacking constraint adopted in three-level TD-based multitoning algorithms introduces a bias to favor the assignment of black dots in their outputs as the brighter energy layer is handled prior to the darker energy layer, which makes them difficult to preserve the bright spatial image features in their outputs. In Chapter 3, we proposed to process the two energy layers in an interleaving manner to reduce the bias. However, it still cannot solve the problems from the root as not all energy layers are taken into account when processing a particular energy layer. The binary halftoning processes for individual energy layers cannot be jointly optimized simultaneously. As a result, the halftones produced for individual energy layers can be at most optimized for their associated layers only. As the multitoning output is produced by combining the halftones, its quality cannot be globally optimized.

Obviously, in order to solve these problems, the energy layers should not be processed one by one sequentially. Instead, they should all be taken into account at the same time when determining the output intensity value of a specific pixel in the final multitone. In this chapter, we will show how this can be done with another proposed approach.

This chapter is organized as follows. Section 4.2 presents our proposed method that solves the addressed problems by taking both energy layers into account when assigning a dot. Section 4.3 shows some simulation results for evaluating the performance of the proposed method. Finally, a summary of the chapter is given in Section 4.4.

4.2 TD-based multitoning based on a complex plane

Consider the case that one wants to produce a three-level multitone for a given input image ***A*** with threshold decomposition and the decomposition scheme suggested by Suetake et al. [Suet08] is adopted. In such a case, image ***A*** is decomposed into a brighter energy layer $\boldsymbol{A_1}$ and a darker energy layer $\boldsymbol{A_2}$ as follows.

$$A_1(x,y) = 2A(x,y) - A(x,y)^2 \qquad \text{for all } (x,y) \qquad (4.1)$$

$$A_2(x,y) = A(x,y)^2 \qquad \text{for all } (x,y) \qquad (4.2)$$

Two binary halftones, say $\boldsymbol{B_1}$ and $\boldsymbol{B_2}$, are then produced based on energy layers $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ subject to a stacking constraint and a three-level multitone ***B*** is obtained by averaging them as

$$B(x,y) = \frac{B_1(x,y) + B_2(x,y)}{2} \qquad \text{for all } (x,y) \qquad (4.3)$$

Note that any binary halftoning algorithm can be applied to produce $\boldsymbol{B_1}$ and $\boldsymbol{B_2}$, but the selection of the algorithm affects the quality of the final multitoning result.

In the proposed framework, the Three-level multitoning process is considered as a process which puts the white dots and black dots onto a gray substrate. For each pixel in the multitone, its intensity value, say $B(x,y)$, is determined by evaluating both $A_1(x,y)$ and $A_2(x,y)$ simultaneously at the time the pixel is selected to assign a value. In such an arrangement, there is no bias to favor either the white or black dots and hence the aforementioned problems can be resolved automatically.

To start the process, one has to estimate the total budgets of black dots and white dots that should be put on the gray substrate. The budgets can be determined based on the working principle of conventional TD-based multitoning algorithms as follows. According to the rules, the white and black dot budgets, which are defined as the expected total numbers of white and black pixels, for the binary halftones $\boldsymbol{B_1}$ and $\boldsymbol{B_2}$ should be given by

$$\textit{white dot budget for } B_d = \sum_{(x,y)} A_d(x,y)$$

$$\textit{black dot budget for } B_d = \sum_{(x,y)} \{1 - A_d(x,y)\} \qquad \text{for } d=1,2 \qquad (4.4)$$

The stacking constraint requires that $B_2(x,y)=0$ whenever $B_1(x,y)=0$. Under this constraint and the connection given by Eqn. (4.3), $B_1(x,y)=0$ is the necessary and sufficient condition for $B(x,y)=0$. Hence, the black dot budget for the three-level multitone $\boldsymbol{B}$ is the same as the black dot budget for $\boldsymbol{B_1}$ and is given by

$$\textit{black dot budget for } \boldsymbol{B},\ D_b = \sum_{(x,y)} \{1 - A_1(x,y)\} \qquad (4.5)$$

Similarly, the connection given by Eqn. (4.3) and the fact that $B_2(x,y)=1$ can only happen when $B_1(x,y)=1$ make $B_2(x,y)=1$ be the necessary and sufficient condition for $B(x,y)=1$. Accordingly, the white dot budget for the three-level multitone $\boldsymbol{B}$ is the same as the white dot budget for $\boldsymbol{B_2}$ and is given by

$$\textit{white dot budget for } \boldsymbol{B},\ D_w = \sum_{(x,y)} A_2(x,y) \qquad (4.6)$$

As a gray substrate, $B(x,y)$ are initialized to be 0.5 for all (x,y). With the dot budgets on hand, the pixels in $\boldsymbol{B}$ are selected and assigned white or black dots by changing the values of $B(x,y)$ to 1 or 0 one by one until all white and black dot budgets are used up. The implementation is basically a two-step iterative process adapted from the one used in FMED [Chan04].

The first step of the iterative process is to select a pixel to put a dot. The selection is based on a complex energy plane $\boldsymbol{E}$ which is initialized by combining energy layers $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ as follows.

$$E(x,y) = A_2(x,y) + j(1 - A_1(x,y)) \qquad \text{for all } (x,y) \tag{4.7}$$

where $j = \sqrt{-1}$. Note that $A_2(x,y)$ and $(1-A_1(x,y))$, respectively, reflect the likeliness that a white dot should be assigned to $B_2(x,y)$ and the likeliness that a black dot should be assigned to $B_1(x,y)$. Accordingly, $E(x,y)$ can be used to estimate the likeliness that a dot should be put onto the gray substrate. At each iteration, after placing a dot, the complex energy plane $E(x,y)$ is updated to guide us to select the most currently critical pixel in $\boldsymbol{B}$ to place a dot.

Starting with the energy plane $\boldsymbol{E}$ as the region of interest, we repeatedly divide the region of interest into nine overlapped sub-regions of equal size and select the sub-region whose cost is maximum to be the new region of interest. Note that any measure can be applied to evaluate the cost of a sub-region, but it affects the quality of the final multitoning result. Two cost measures are suggested as follows.

$$J_R = \|\max(Real(J), 0) + j \max(Imag(J), 0)\|_2 \tag{4.8}$$

or

$$J_R = \|\max(Real(J), 0) + j \max(Imag(J), 0)\|_1 \tag{4.9}$$

where $J = \sum_{(u,v)\in\Lambda} E(u,v) \cdot M(u,v)$ and Λ denotes the set of pixels in the region and $M(u,v)$ is a mask defined as

$$M(u,v) = \begin{cases} 1 & if \;\; B(u,v) = 0.5 \\ 0 & else \end{cases} \tag{4.10}$$

Note that the physical meaning of $B(u,v)$=0.5 is that pixel (u,v) in $\boldsymbol{B}$ has not yet been selected to put either a white or black dot. We repeat the above procedures to update the region of interest until a pixel location is reached.

In the conventional TD framework, $A_2(x,y)$ and $(1- A_1(x, y))$, respectively, reflect the appropriateness of assigning 1 (a white dot) to $B_2(x,y)$ and 0 (a black dot) to $B_1(x,y)$. The closer to 0 their values, the less appropriate the assignments are. In other words, the larger the value of $(1- A_1(x, y))^2 + A_2(x, y)^2$, the more appropriate it is to make either $B_2(x,y)$=1 or $B_1(x,y)$=0. As $B_2(x,y)$=1 and $B_1(x,y)$=0 lead to $B(x,y)$=1 and $B(x,y)$=0, respectively, it implies that $B(x,y)$ should no longer be 0.5. Adjusting the initial value of $B(x,y)$ is equivalent to putting a dot (either black or white) onto the gray substrate at location (x,y). The searching procedures in this step guide us to search for the most appropriate location to put a dot.

The second step is to assign an appropriate value to the pixel selected in the first step and update energy plane $\boldsymbol{E}$ accordingly. Let the location of the selected pixel be (p,q). The intensity value of the dot assigned to $B(p,q)$ is determined by

$$B(p,q) = \begin{cases} 1 & if\ A_2(p,q) > 1 - A_1(p,q)\ and\ D_w > 0 \\ 0 & else \end{cases} \tag{4.11}$$

After the dot assignment, the corresponding dot budget for $\boldsymbol{B}$ is reduced by 1.

Based on Eqn. (4.3), we have $B_1(p,q) = B_2(p,q) = B(p,q)$. The energy layers $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ are then updated accordingly to update the complex energy plane $\boldsymbol{E}$ for selecting another pixel in $\boldsymbol{B}$ to put a dot in next iteration. In particular, for each energy layer $\boldsymbol{A_d}$, where $d\in\{1,2\}$, the difference between $B_d(p,q)$ and $A_d(p,q)$ is diffused to $A_d(p,q)$'s neighbors to update the energy layer $\boldsymbol{A_d}$ as follows.

$$A'_d(x,y) = \begin{cases} 0 & if\ (x,y) = (p,q) \\ A_d(x,y) - w_{x-p,y-q} \cdot \left(B_d(p,q) - A_d(x,y)\right) \cdot \frac{M(x,y)}{S} & else \end{cases} \tag{4.12}$$

where $A'_d(x,y)$ and $A_d(x,y)$ are, respectively, the values of pixel (x,y) in energy layer $\boldsymbol{A_d}$ after and before the error diffusion process, $B_d(p,q)$ is the value assigned to pixel (p,q), $w_{s,t}$ for $(s,t) \in \Omega$ is a filter weight of a non-causal diffusion filter with support Ω, and

$$S = \sum_{(x-p,y-q)\in\Omega}\left(w_{x-p,y-q} \cdot M(x,y)\right) \quad (4.13)$$

The two steps are repeated until all the black and white dot budgets for $\boldsymbol{B}$ are exhausted. Since both steps are carried out based on the complex energy plane $\boldsymbol{E}$ and the whole iterative process complies with the framework of FMED, the process is referred to as complex plane MED in this thesis.

Figure 4.1 shows the operational flow of the proposed three-level TD-based multitoning method. For reference purpose, the proposed multitoning algorithm is referred to as TD-CMED in this thesis.

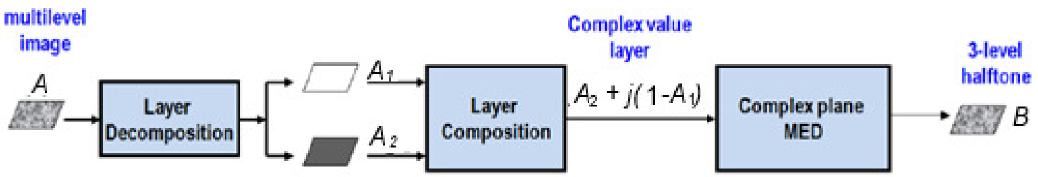


Figure 4.1 Operation flow of TD-CMED algorithm for producing three-level halftone

4.3 Simulation results and discussion

Simulation is carried out to evaluate the performance of the proposed method on a set of nine 256-level testing images [Spri] of size 512×512 each. In its realization, a 5×5 noncausal diffusion filter with filter coefficients

$$w_{s,t} = \begin{cases} \frac{1}{\sqrt{s^2+t^2}} & if\ |s|,|t| \le D = 2\ and\ |s|+|t| \ne 0 \\ 0 & else \end{cases} \quad (4.14)$$

is used as the default diffusion filter in the simulation. However, there is no limitation on the default diffusion filter used in the proposed algorithm. When the number of not-yet-processed pixels in the filter support is 0 (i.e. $S=0$), the value of D is increased by 1 gradually to increase the filter support until $S\neq 0$.

As mentioned in Section 4.2, in TD-CMED various cost measures such as those suggested in Eqns. (4.8) and (4.9) can be applied to evaluate the cost of a sub-region when selecting a new sub-region as the new region of interest to search the most critical pixel to put a dot. Table 4.1 shows the performance of TD-CMED in terms of Mean Structural Similarity Index (MSSIM) [Wang04] when different cost measures are used.

MSSIM is an improved version of Universal Objective Image Quality Index (UQI) [Wang02] which can be used to measure the information loss after multitoning. The MSSIM measurement is sensitive to structural information degradation which reflects the difference between the original image and its multitoning results. The higher the value of MSSIM is, the closer to the original a multitoning output is, which implies a better output quality. Table 4.1 shows that the cost measure using Eqn. (4.8) is better than using cost measure Eqn. (4.9) to work with TD-CMED.

Table 4.1 MSSIM PERFORMANCE OF TD-CMED WHEN DIFFERENT COST MEASURES ARE USED TO LOCATE A PIXEL TO PUT A DOT

Image	**Cost Measure**	
	Eqn. (4.8)	**Eqn. (4.9)**
Peppers	0.0963	0.0918
Man	0.1409	0.1316
Boat	0.1482	0.1398
Lena	0.1076	0.1008
Goldhill	0.1375	0.1256
Girl	0.0702	0.0643
Barbara	0.2129	0.2052
Baboon	0.3140	0.2986
Airplane	0.1265	0.1201
average	**0.1505**	**0.1420**

Figure 4.2 shows some three-level halftoning results obtained with TD-CMED for visual comparison. One can see that TD-CMED provides the outputs of similar visual quality when Eqns. (4.8) and (4.9) are used as the cost measures. Based on the simulation results in Table 4.1 and Figure 4.2, we suggest using Eqn. (4.8) as the cost measure when using TD-CMED to produce a three-level multitone. Hereafter, when we refer to TD-CMED, we are actually referring to the scheme where Eqn. (4.8) is used as the cost measure to locate a pixel to put a dot.

Table 4.2 shows the MSSIM performances of various three-level TD-based multitoning algorithms. It shows that the proposed algorithm outperforms the others in terms of this measure.

Figures 4.3 and 4.4 show some three-level multitoning results obtained with different three-level TD-based multitoning algorithms for visual comparison. One can see that TD-CMED can render the texture (e.g. the patterns on the tablecloth, the scarf and the trousers etc.) in a way better than the others. Both the bright and dark spatial features are reported faithfully in the output of TD-CMED. Similar observations can be found in the simulation results of other testing images shown in the Appendix. Figure 4.5 shows the multitones obtained with various algorithms for a ramp input.

As complex number operations are required in TD-CMED, its realization complexity is higher than TD-FMEDi's. In our Matlab simulation, the running time of TD-CMED roughly doubles that of TD-FMEDi.

(a)

(b) (c)

Figure 4.2 Simulation results of TD-CMED when different cost measures are used to search a pixel to put a dot: (a) Original image, (b) Using Eqn. (4.8) and (c) Using Eqn. (4.9).

Table 4.2 MSSIM PERFORMANCES OF VARIOUS THREE-LEVEL MULTITONING ALGORITHMS

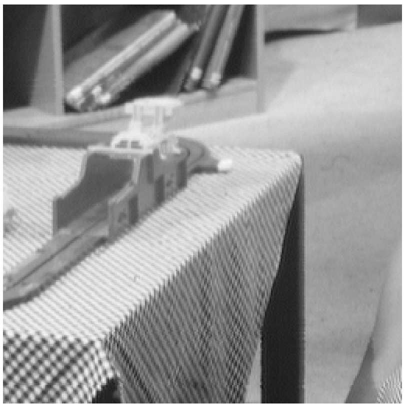
Image	[Suet08]	[Rodr08]	[Fung10]	TD-FMEDi	TD-CMED
Peppers	0.0652	0.0624	0.0825	0.0843	0.0963
Man	0.0772	0.0717	0.1123	0.1138	0.1409
Boat	0.0842	0.0770	0.1178	0.1175	0.1482
Lena	0.0601	0.0560	0.0879	0.0885	0.1076
Goldhill	0.0667	0.0612	0.1079	0.1079	0.1375
Girl	0.0366	0.0348	0.0578	0.0586	0.0702
Barbara	0.1346	0.1229	0.1803	0.1834	0.2129
Baboon	0.1805	0.1701	0.2715	0.2721	0.3140
Airplane	0.0760	0.0737	0.1073	0.1015	0.1265
Average	**0.0868**	**0.0811**	**0.1250**	**0.1253**	**0.1504**

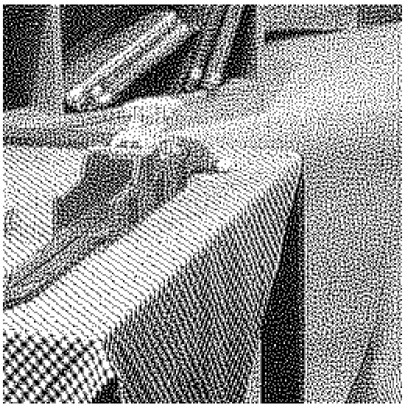

Original TD-CMED

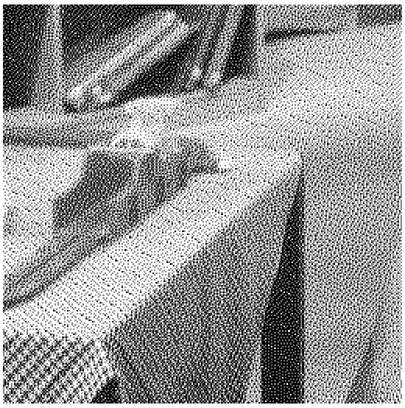

Suetake et al.[Suet08] Rodriguez et al. [Rodr08]

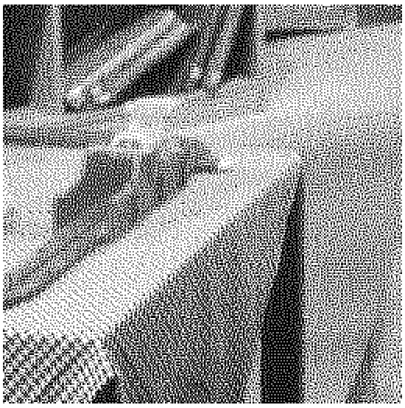

Fung et al. [Fung10] TD-FMEDi

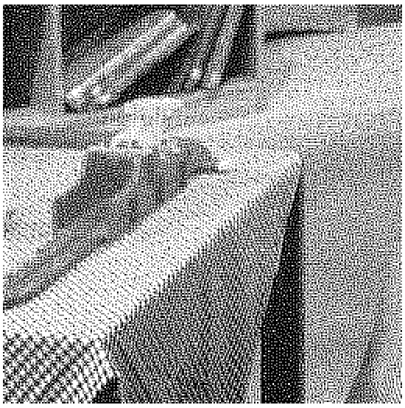
Figure 4.3 Simulation results on Barbara for different three-level multitoning algorithms – part I


Original


TD-CMED


Suetake et al.[Suet08]


Rodriguez et al. [Rodr08]


Fung et al. [Fung10]

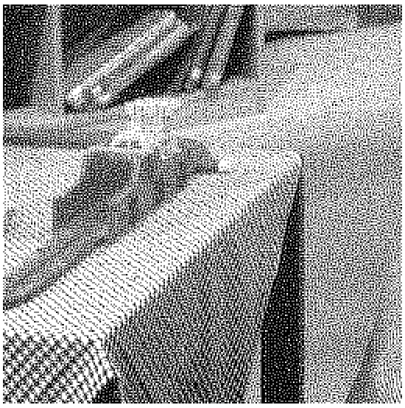
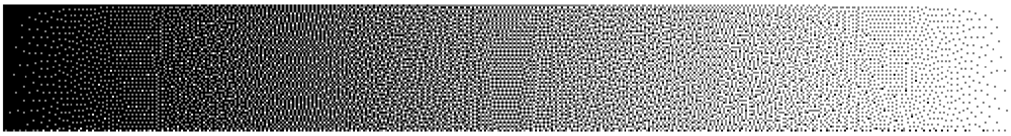
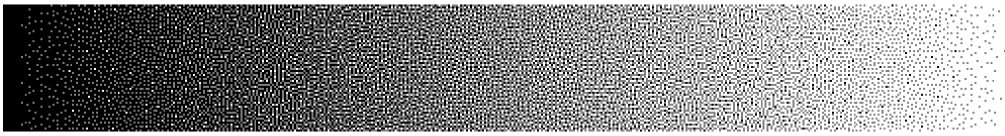

TD-FMEDi

Figure 4.4 Simulation results on Barbara of different three-level multitoning algorithms – part II

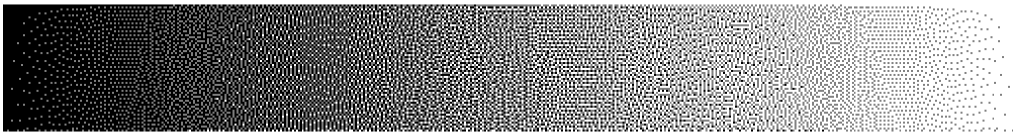
Original



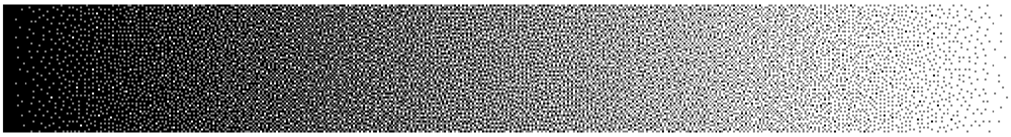
Suetake et al. [Suet08]



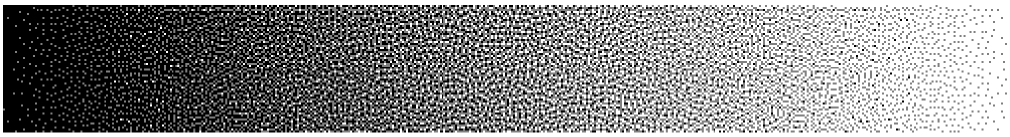
Fung et al. [Fung10]



Rodríguez et al. [Rodr08]



TD-FMEDi



TD-CMED

Figure 4.5 Three-level multitoning outputs of ramp image

4.4 Summary

Conventional three-level TD-based multitoning algorithms process the brighter layer first, so the bright spatial features are more difficult to preserve in their outputs. In this chapter, we have proposed a solution to handle all energy layers simultaneously to eliminate the bias. Both the dark and bright spatial features on the input images can then be faithfully preserved in their multitoning outputs. Simulation results have showed that the proposed method can give a better result than the conventional TD-based multilevel halftoning algorithms [Suet08, Rodr08, Fung10] in both the subjective and objective measures. It is also better than the algorithm proposed in Chapter 3 as it can effectively eliminates the bias from the root.

Chapter 5

Feature-preserving TD-based Multitoning

5.1 Introduction

In Chapters 3 and 4, we proposed two solutions to remove the bias introduced by the stacking constraint used in the TD-based multitoning algorithms such that we can produce three-level multitones of better quality. As a preliminary study of the issue, the number of the output levels of the target multitones is limited to 3 in these chapters. In Chapter 5, we extend the idea in Chapter 3 and propose a multilevel halftoning algorithm to produce multitones for any number of output levels.

Consider the case to produce a *m*-level multitone based on the TD-based framework shown in Figure 2.6. Under the TD-framework, an input image should be decomposed into *m*-1 layers. When *m* is larger than 3, it may not be possible for us to handle all the layers simultaneously as suggested in Chapter 4. In other words, some layers are unavoidably selected to be processed earlier.

An observation we have had is that, in practice, human is more sensitive to the dots of higher contrast. Hence, in the multilevel halftoning output, the relative locations among the dots of higher contrast should be more critical. From this point of view, at any particular moment during the multilevel halftoning process, we should handle the layers having the maximum energy level and the minimum energy level simultaneously as, under the stacking constraint, the locations of the brightest and the darkest dots in the final multilevel halftone are determined by the binary halftoning results of these two layers. In other words, we should handle layers $\boldsymbol{A_1}$ and $\boldsymbol{A_{m-1}}$ first to position the dots of minimum intensity and maximum intensity, then layers $\boldsymbol{A_2}$ and $\boldsymbol{A_{m-2}}$ to position the dots of second minimum intensity and second maximum intensity, and so on. This process should be repeated until all layers are handled to reach the final multilevel halftoning output.

This chapter is organized as follows. This section gives an introduction of the idea to develop the multitoning algorithm presented in this chapter. In Section 5.2 we show how the idea presented in Chapter 3 can be extended to preserve the image features in a *m*-level multitoning output. Simulation results are shown in Section 5.3 for evaluating the performances of the proposed multitoning algorithm and a summary of the chapter is given in Section 5.4.

5.2 Proposed algorithm

Let us recall that Figure 2.6 shows the framework of a conventional TD-based multitoning algorithm that decomposes an input image into *m*-1 layers for processing and produces a *m*-level multitone. In particular, the layers are processed with a binary halftoning algorithm one by one under a stacking constraint.

In theory, there is no limitation on selecting the binary halftoning algorithm used to halftone a layer when threshold decomposition is used. As FMED [Chan04] has been proved to be effective in preserving features and it is more flexible to introduce dots of different nature to the outputs of individual layers, naturally it is selected to halftone the layers in our proposed multitoning algorithm. To be more precise, we adopt the general idea of FMED and apply it to our proposed multitoning algorithm with modification.

Figure 5.1 shows the operation flow of the proposed algorithm for producing a *m*-level output, where *m* is a positive odd integer. The decomposition scheme suggested by Suetake et al. [Suet08], which is formulated as in Eqn. (2.2), is exploited here to separate an input image $\boldsymbol{A}$ into several layer $\boldsymbol{A}_d$ first. Then, the whole multilevel halftoning process is split into $(m\text{-}1)/2$ stages. In the first stage, layers $\boldsymbol{A}_1$ and $\boldsymbol{A}_{m-1}$ are processed to locate the darkest and the brightest pixels in the final multilevel halftone. To achieve this, we alternately locate a pixel in layer $\boldsymbol{A}_{m-1}$ to assign a "1" and locate a pixel in layer $\boldsymbol{A}_1$ to assign a "0". Under the stacking constraint, this is equivalent to alternately positioning the brightest and the darkest dots in the final multilevel halftoning output of the input image.

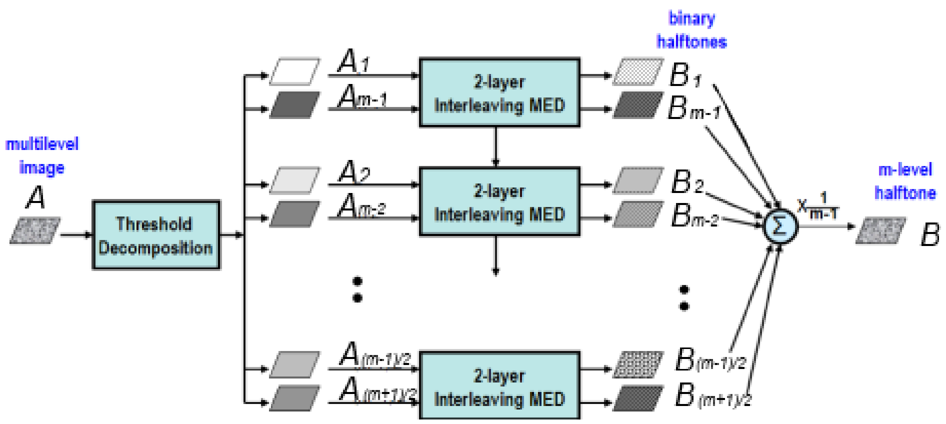


Figure 5.1 Operation flow of g-TD-FMEDi algorithm for producing m-level halftone

Though layers $\boldsymbol{A}_n$ for n = 2, 3,… m-2 do not contribute to this stage of the process, they are affected by the processing result of this stage. According to the stacking constraint, when a pixel in the brightest layer is selected to put a black dot in its binary halftone, a black dot must also be assigned to the corresponding pixels in the binary halftones of all lower energy layers to maintain consistency. Based on the same philosophy, when a pixel in the darkest layer is selected to put a white dot in its binary halftone, a white dot must also be assigned to the corresponding pixels in the binary halftones of all higher energy layers. Hence, the corresponding pixels in layers other than $\boldsymbol{A}_1$ and $\boldsymbol{A}_{m-1}$ should also be assigned a corresponding value when a pixel is selected in either layer $\boldsymbol{A_1}$ or $\boldsymbol{A_{m-1}}$ to put a dot. For each one of the affected layers, the quantization error in the selected pixel is diffused to its neighbors in the same layer.

After stage 1, the binary halftones of layers $\boldsymbol{A_1}$ and $\boldsymbol{A_{m-1}}$ are finalized and they will not be affected in the subsequent stages of the proposed multilevel halftoning algorithm. Layers $\boldsymbol{A_2}$ and $\boldsymbol{A_{m-2}}$ now, respectively, become the brightest and the darkest layers among those which have not yet been processed and hence they are the two most critical layers. Accordingly, they are processed in the second stage. Based on the same philosophy, the layers are paired up to be processed in subsequent stages until all of them are processed to produce m binary halftones. In general, we process layers $\boldsymbol{A_n}$ and $\boldsymbol{A_{m-n}}$ in stage n for n = 1, 2, … (m-1)/2.

Suppose we are now processing layers $\boldsymbol{A_n}$ and $\boldsymbol{A_{m-n}}$ in stage n, where $n \leq (m-1)/2$. At the beginning of the stage, we have to estimate the white dot budget for the lower energy layer $\boldsymbol{A_{m-n}}$ and the black dot budget for the higher energy layer $\boldsymbol{A_n}$ as follows.

$$\text{Budget1} = \sum_{(x,y)} A_{m-n}(x,y)$$

$$\text{Budget0} = N_o - \sum_{(x,y)} A_n(x,y) \tag{5.1}$$

where N_o is the total number of the pixels that have not yet been assigned a value in the binary output plane of layer $\boldsymbol{A_n}$. A budget ratio is then determined as

$$\text{Budget ratio} = \frac{\sum_{(x,y)} A_{m-n}(x,y)}{N_o - \sum_{(x,y)} A_n(x,y)} \tag{5.2}$$

To achieve a balance on preserving the dark and bright spatial features in an image, we remove the bias in either white or black dots by alternately positioning them under the guidance of the budget ratio. When the dots are put in the layers, the black dot budget for layer $\boldsymbol{A_n}$ and the white dot budget for layer $\boldsymbol{A_{m-n}}$ are consumed accordingly. At the time when we determine which type of dots should be positioned next, we can pick the one that minimizes the difference between the budget ratio and the ratio of the remaining white dot budget to the remaining black dot budget.

The search of the pixel to put a dot is based on an energy plane $\boldsymbol{E}$. If a white dot is desired, the energy plane $\boldsymbol{E}$ is initialized to be layer $\boldsymbol{A_{m-n}}$. Otherwise it is initialized to be layer $\boldsymbol{A_n}$. The search is carried out as follows. Starting with the selected energy plane $\boldsymbol{E}$ as the region of interest, we repeatedly divide the region of interest into nine overlapped sub-regions of equal size and pick a sub-region among them to be the new region of interest based on a selection criterion. When we are looking for a dark pixel in layers $\boldsymbol{A_n}$, minimum intensity guidance is adopted and we pick the one having the smallest sum of its pixel values. Otherwise, maximum intensity guidance is adopted and the one having the largest sum is picked. We repeat the above steps to update the region of interest until a pixel location is reached.

Without loss of generality, let $\boldsymbol{A_k}$ be the layer that we have located a pixel to put a dot and the location of the pixel is (p,q). A dot with the corresponding intensity level is then put at $B_k(p,q)$, where $\boldsymbol{B_k}$ is the binary output plane of layer $\boldsymbol{A_k}$. The difference between $B_k(p,q)$ and $A_k(p,q)$ is then diffused to $A_k(p,q)$'s neighbors to update layer $\boldsymbol{A_k}$. In particular, layer $\boldsymbol{A_k}$ is updated with the diffusion equation defined as follows.

$$A'_k(x,y) = \begin{cases} 0 & if\ (x,y) = (p,q) \\ A_k(x,y) - w_{x-p,y-q} \cdot \left(B_k(p,q) - A_k(x,y)\right) \cdot \frac{R(x,y)}{S} & else \end{cases} \quad (5.3)$$

where $A_k(x,y)$ and $A'_k(x,y)$ are, respectively, the values of pixel (x,y) in layer k before and after the error diffusion process, $B_k(p,q)$ is the value assigned to pixel (p,q), $R(x,y)$ is a mask defined as

$$R(x,y) = \begin{cases} 0 & \text{if } B_k(x,y) \text{ has been assigned a dot} \\ 1 & \text{else} \end{cases} \tag{5.4}$$

$w_{s,t}$ for $(s,t) \in \Omega$ is a filter weight of a noncausal diffusion filter with support Ω, and

$$S = \sum_{(x-p,y-q)\in\Omega} (w_{x-p,y-q} \cdot R(x,y)) \tag{5.5}$$

Under the stacking constraint, some other layers will also be affected and they should also be updated. In particular, we have

$$B_i(p,q) = 1 \quad for\ m-k \le i \le k \quad if\ k \in \{\tfrac{m+1}{2}, \tfrac{m+3}{2}, \dots m-1\} \tag{5.6}$$

or

$$B_i(p,q) = 0 \quad for\ m-k \ge i \ge k \quad if\ k \in \left\{1,2, \dots \tfrac{m-1}{2}\right\} \tag{5.7}$$

For each affected layer $\boldsymbol{A_i}$, we diffuse the error at pixel (p,q), which is the difference between $B_i(p,q)$ and $A_i(p,q)$, to $A_i(p,q)$'s neighbors with Eqn. (5.3) to update layer $\boldsymbol{A_i}$ as well.

Dots are consumed one by one and the layers are updated as mentioned above until all white dot budget for the lower energy layer $\boldsymbol{A_{m-n}}$ and all black dot budget for the higher energy layer $\boldsymbol{A_n}$ are used up. The pixels whose values in $\boldsymbol{B_{m-n}}$ and $\boldsymbol{B_n}$ are left unassigned are backfilled as follows to complete the construction of the binary halftones of layers $\boldsymbol{A_{m-n}}$ and $\boldsymbol{A_n}$.

$$B_n(x, y) = 1 \qquad \text{for all } (x,y) \text{ in } \boldsymbol{B_n} \text{ not assigned } 0 \qquad (5.8)$$

and

$$B_{m-n}(x, y) = 0 \qquad \text{for all } (x,y) \text{ in } \boldsymbol{B_{m\text{-}n}} \text{ not assigned } 1 \qquad (5.9)$$

This backfill process does not affect any other layers and it concludes the operation of this stage.

Layers are handled two by two until all of them are processed to produce their binary halftones. The multilevel halftone is then obtained by combining their binary halftones as follows.

$$B(x, y) = \frac{1}{m-1} \sum\nolimits_{n=1,2\ldots m-1} B_n(x, y) \qquad (5.10)$$

Figure 5.2 shows the flow of the proposed multilevel halftoning algorithm in a form of pseudo code. It summarizes the operation of the algorithm proposed in this section. By comparing it with the operation flow of the three-level multitoning algorithm proposed in Chapter 3 (i.e TD-FMEDi), one can see that the algorithm proposed in this chapter is actually a generalized version of TD-FMEDi and TD-FMEDi is only a special case of m = 3. For reference purpose, the algorithm proposed in this chapter is referred to as generalized TD-FMEDi (g-TD-FMEDi).

As a remark, we note that, though the pseudo code provided in Figure 5.2 is for producing an odd-level multitone, it can be easily modified to handle the case for producing an even-level multitone based on the same idea presented in this chapter.

Decompose input image $\boldsymbol{A}$ to m–1 layers

Initialize mask $\boldsymbol{R}(x,y)$=1 for all (x,y) to indicate they are not processed

FOR layer pair n=1,2,...(m–1)/2

Budget1 = $round\left(\sum_{(x,y)} \boldsymbol{A}_{m-n}(x,y)\right)$

Budget0 = $round\left(N_0 - \sum_{(x,y)} \boldsymbol{A}_n(x,y)\right)$

BudgetRatio = Budget1/Budget0;

WHILE Budget1+Budget0>0

IF Budget1 ≥ BudgetRatio × Budget0

Energy plane $\boldsymbol{E} = \boldsymbol{A}_{m-n}$

Search for a pixel location in $\boldsymbol{E}$ via maximum intensity guidance

FOR i = 1 to m–n

$\boldsymbol{B}_i(x_o,y_o)$ = 1 % Assume that the location is (x_o,y_o)

Diffuse error of $\boldsymbol{A}_i(x_o,y_o)$ to its neighbors in $\boldsymbol{A}_i$

END

Budget1 = Budget1 - 1

ELSE

Energy plane $\boldsymbol{E} = \boldsymbol{A}_n$

Search for a pixel location in $\boldsymbol{E}$ via minimum intensity guidance

FOR i = n to m–1

$\boldsymbol{B}_i(x_o,y_o)$ = 0 % Assume that the location is (x_o,y_o)

Diffuse error of $\boldsymbol{A}_i(x_o,y_o)$ to its neighbors in $\boldsymbol{A}_i$

END

Budget0 = Budget0 - 1

END

Update mask $\boldsymbol{R}(x_o,y_o)$=0 to indicate pixel (x_o,y_o) is processed

END

$\boldsymbol{B}_n(x,y)$ = 1 for all (x,y) in $\boldsymbol{B}_n$ not assigned 0

$\boldsymbol{B}_{m-n}(x,y)$ = 0 for all (x,y) in $\boldsymbol{B}_{m-n}$ not assigned 1

END

Multilevel halftone is $\boldsymbol{B}(x,y) = \frac{1}{m-1}\sum_{n=1,2,\ldots m-1} \boldsymbol{B}_n(x,y)$

Figure 5.2 Pseudo code of g-TD-FMEDi algorithm

5.3 Simulation results and discussion

Simulation was carried out to evaluate the performance of the proposed multitoning algorithm with a set of standard 256 gray-level testing images [Spri] shown in Appendix. All testing images are of size 512×512. For comparison, the performances of two additional conventional TD-based multitoning algorithms ([Suet08] and [Fung10]) were also evaluated. In particular, Suetake et al.'s algorithm [Suet08] halftones each layer with SED using Floyd Steinberg's diffusion filter with serpentine scanning, while Fung et al.'s algorithm [Fung10] halftones each layer with FMED [Chan04]. All evaluated algorithms adopt the same decomposition scheme specified in Eqn. (2.2).

As mentioned earlier, TD-FMEDi is a special case of g-TD-FMEDi in which the number of output levels is 3. The performance of g-TD-FMEDi in producing three-level multitones can hence be evaluated based on the simulation results presented in Section 3.3. In this section, we will put our focus on the performance in producing multitones of more output levels.

Tables 5.1 and 5.2 show the MSSIM performances of various algorithms when they are used to produce five-level and seven-level multitones. In terms of MSSIM, g-TD-FMEDi performs better as compared with the other evaluated algorithms.

Table 5.1 MSSIM PERFORMANCES OF VARIOUS FIVE-LEVEL MULTITONING ALGORITHMS

Image	[Suet08]	[Fung10]	[Sara12]	g-TD-FMEDi
Peppers	0.1086	0.1351	0.2183	0.1388
Man	0.1345	0.1843	0.4034	0.1874
Boat	0.1439	0.1909	0.3727	0.1947
Lena	0.1044	0.1460	0.3518	0.1486
Goldhill	0.1211	0.1842	0.4238	0.1876
Girl	0.0690	0.1009	0.3113	0.1067
Barbara	0.2144	0.2704	0.4649	0.2749
Baboon	0.2954	0.3963	0.6269	0.4003
Airplane	0.1256	0.1636	0.3144	0.1627
Average	**0.1463**	**0.1969**	**0.3875**	**0.2002**

Table 5.2 MSSIM PERFORMANCES OF VARIOUS SEVEN-LEVEL MULTITONING ALGORITHMS

Image	[Suet08]	[Fung10]	[Sara12]	g-TD-FMEDi
Peppers	0.1427	0.1731	0.3282	0.1800
Man	0.1787	0.2357	0.6044	0.2401
Boat	0.1892	0.2427	0.5603	0.2485
Lena	0.1399	0.1886	0.5062	0.1928
Goldhill	0.1670	0.2403	0.5741	0.2447
Girl	0.0979	0.1386	0.4552	0.1450
Barbara	0.2713	0.3297	0.6124	0.3347
Baboon	0.3758	0.4717	0.7549	0.4790
Airplane	0.1643	0.2090	0.4285	0.2067
Average	**0.1919**	**0.2477**	**0.5360**	**0.2524**

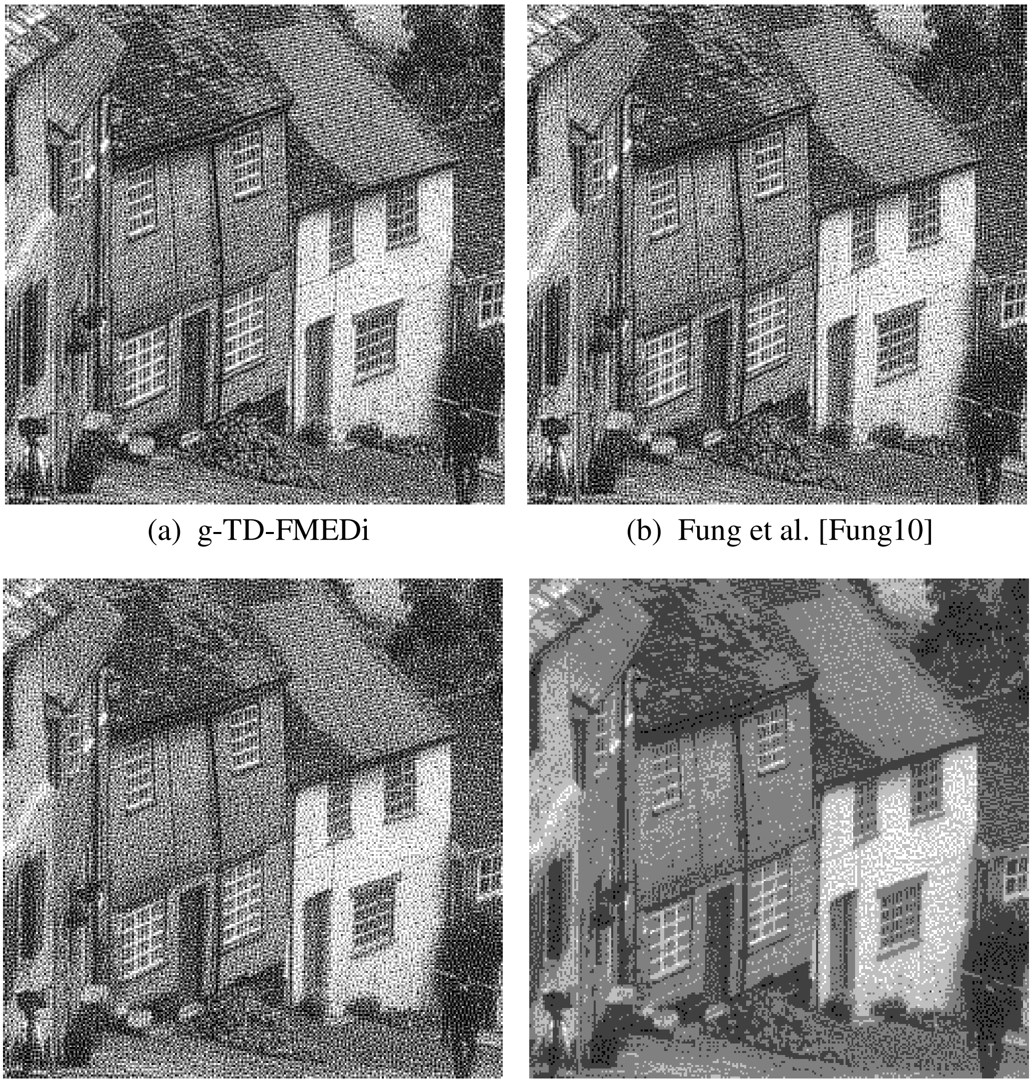
Figures 5.3 and 5.4, respectively, show the five-level and the seven-level halftoning results obtained with different multilevel halftoning algorithms for evaluation. Their original carries a lot of image details and is shown in Figure 3.4(a) as a reference for comparison. In general, for the same multilevel halftoning algorithm, the more the output levels of its produced multitone, the better the quality of the multitone is. By comparing the rooftops, the drain pipes and the window frames in Figures 5.3 and 5.4, one can tell that the proposed algorithm can preserve the spatial details better no matter how many output levels a halftone contains. However, the difference between the output of Fung et al.'s [Fung10] and that of g-TD-FMEDi becomes less significant when there are more output levels. Similar observations can be found in the simulation results of other testing images shown in the Appendix.

Both Fung et al.'s and g-TD-FMEDi are FMED-based multilevel halftoning algorithms and adopt the same threshold decomposition scheme. Their only difference is that Fung et al.'s handles the layers one by one while the proposed algorithm handles them two by two such that the two layers with the largest contrast among those not yet processed can be processed at the same time to remove the bias to either bright or white dots. This makes the proposed algorithm be able to preserve both the bright and dark features in a way better than Fung et al.'s algorithm. Figures 5.5 and 5.6 show, respectively, the five-level and seven-level multitoning results of a combined artificial testing image in which there are only four input gray levels. To a certain extent, both the bright and dark characters in Figures 5.5(b) and 5.6(b) are more recognizable than those in Figures 5.5(c) and 5.6(c).

Figures 5.7 and 5.8 show, respectively, the five-level and seven-level multitoning results of a gray level ramp strip. The output of g-TD-FMEDi offers a smooth transition and there is no banding artifact found in its output. It faithfully preserves the spatial details of an image without purposely emphasizing the high frequency components. In general, a multitone of more output levels provides an output with higher quality.

The complexity of the proposed algorithm is in the order of $O(N_s)$, where N_s is the total number of pixels of the input image. Note that, in the proposed algorithm, once a pixel location of the image is picked, the binary output values of the corresponding locations in all layers are determined and hence the number of layers contributes almost nothing to the complexity. For the conventional threshold decomposition-based algorithms like those in [Suet08], [Rodr08] and [Fung10], the layers are halftoned one by one and hence the complexity is directly proportional to the number of layers. As a final remark, we also note that the complexity of the proposed algorithm can be significantly reduced with the approach proposed in [Lui07] to achieve a real-time implementation.

In our Matlab simulation, the running time of gTD-FMEDi is more or less the same as that of Fung et al.'s algorithm [Fung10]. As compared with the running time of Rodriguez et al.'s [Rodr08] and that of Suetake et al.'s [Suet08], the ratio is 500:1:1 for producing a five-level multitone and 350:1:1 for producing a seven-level multitone. Again, since the realization of all algorithms was not optimized in our simulation, the complexity comparison reported here may not reflect the real situation accurately.



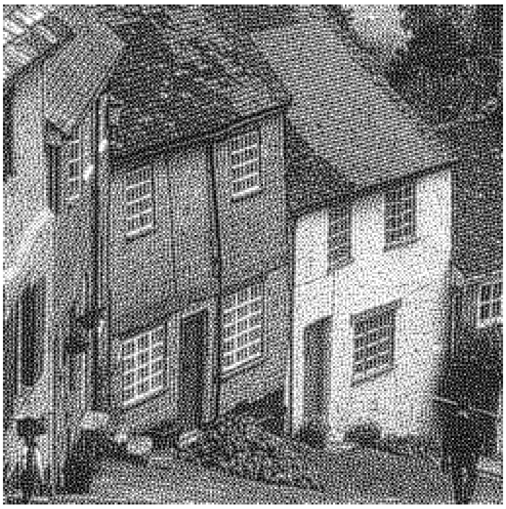
(a) g-TD-FMEDi

(b) Fung et al. [Fung10]

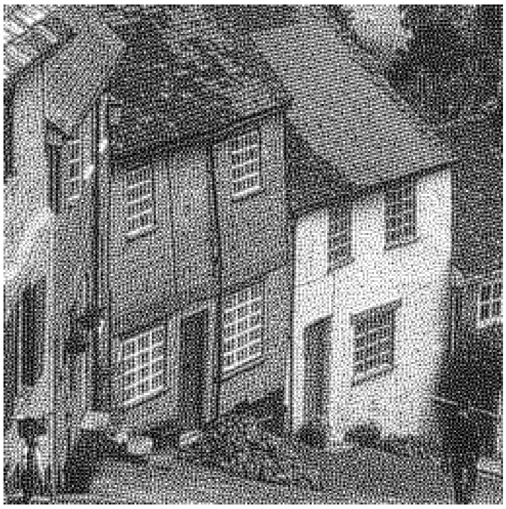
(c) Suetake et al. [Suet08]

(d) Sarailidis et al. [Sara12]

Figure 5.3 Parts of five-level halftoning results of *Goldhill*: (a) g-TD-FMEDi, (b) Fung et al. [Fung10], (c) Suetake et al. [Suet08] and (d) Sarailidis et al. [Sara12]

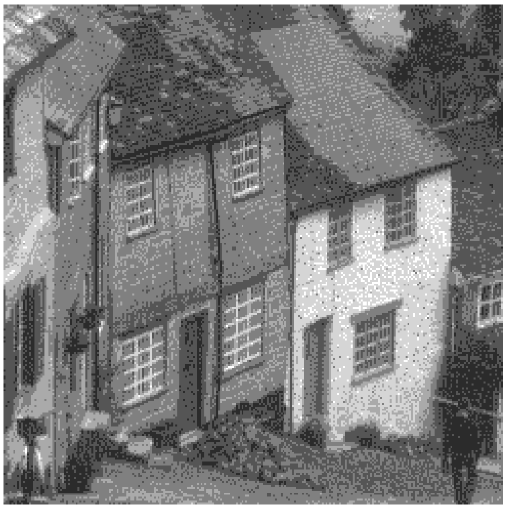


(a) g-TD-FMEDi



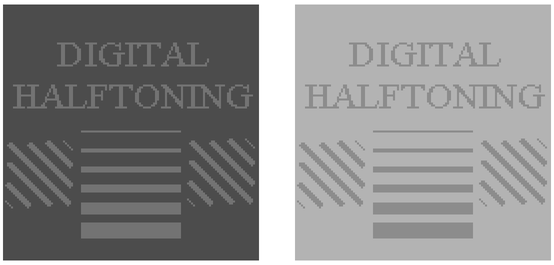
(b) Fung et al. [Fung10]

(c) Suetake et al. [Suet08]

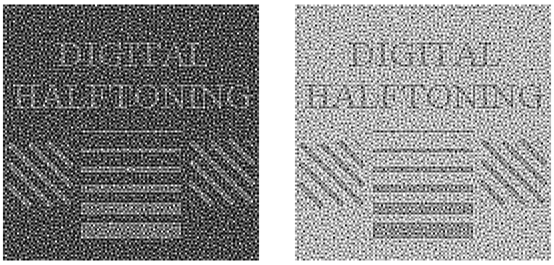


(d) Sarailidis et al. [Sara12]

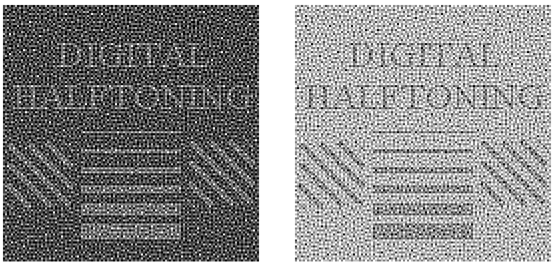
Figure 5.4 Parts of seven-level halftoning results of *Goldhill*: (a) g-TD-FMEDi, (b) Fung et al. [Fung10], (c) Suetake et al. [Suet08] and (d) Sarailidis et al. [Sara12]



(a) Original



(b) g-TD-FMEDi



(c) Fung et al. [Fung10]

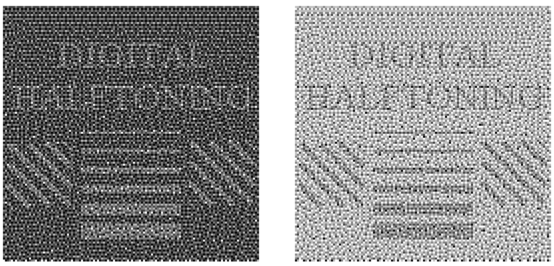
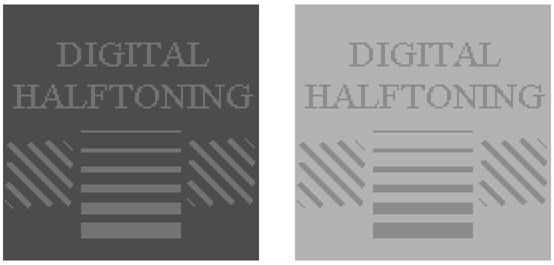
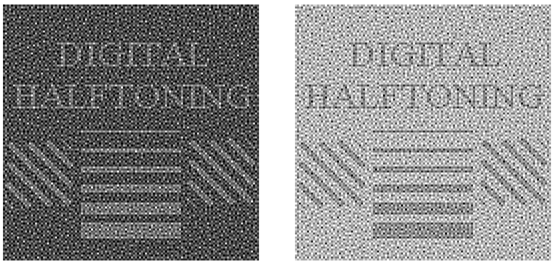

(d) Suetake et al. [Suet08]

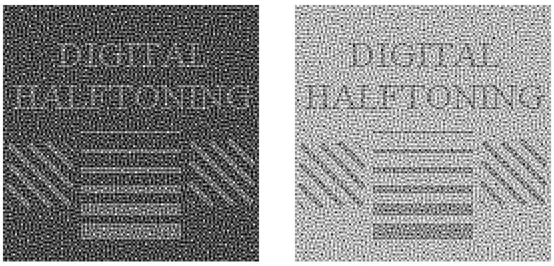
Figure 5.5 Five-level halftoning outputs of artificial testing images



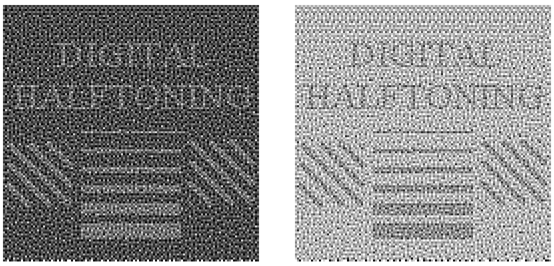
(a) Original



(b) g-TD-FMEDi



(c) Fung et al. [Fung10]



(d) Suetake et al. [Suet08]

Figure 5.6 Seven-level halftoning outputs of artificial testing images

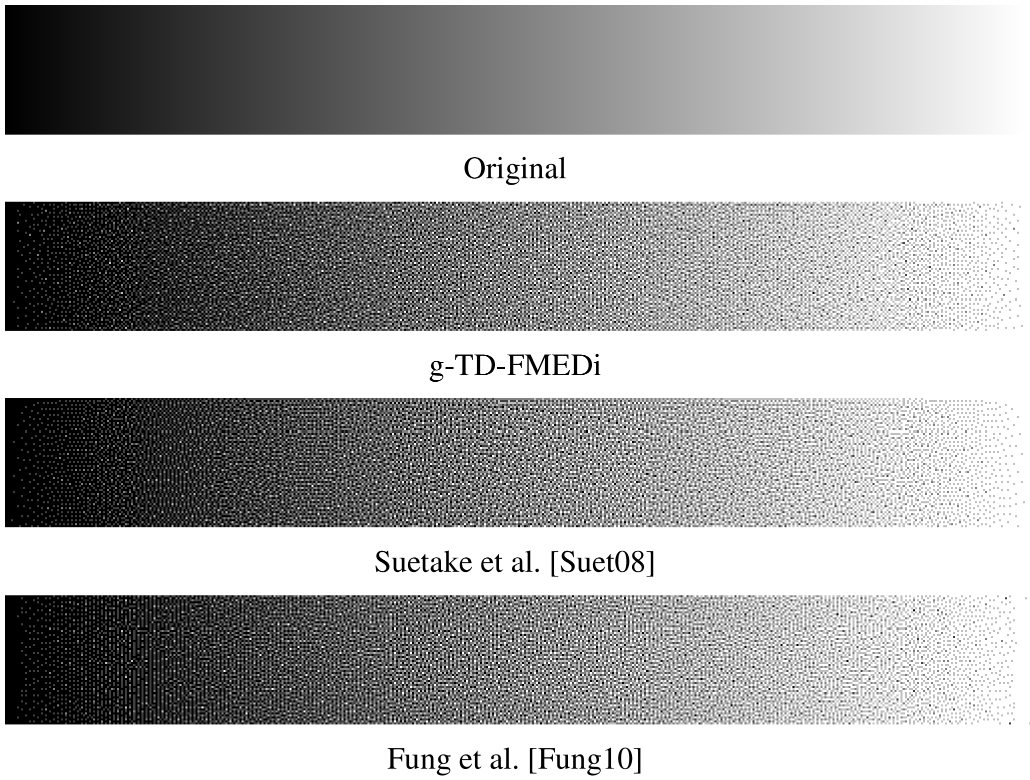


Figure 5.7 Five-level halftoning outputs of gray-scale ramp obtained with different algorithms

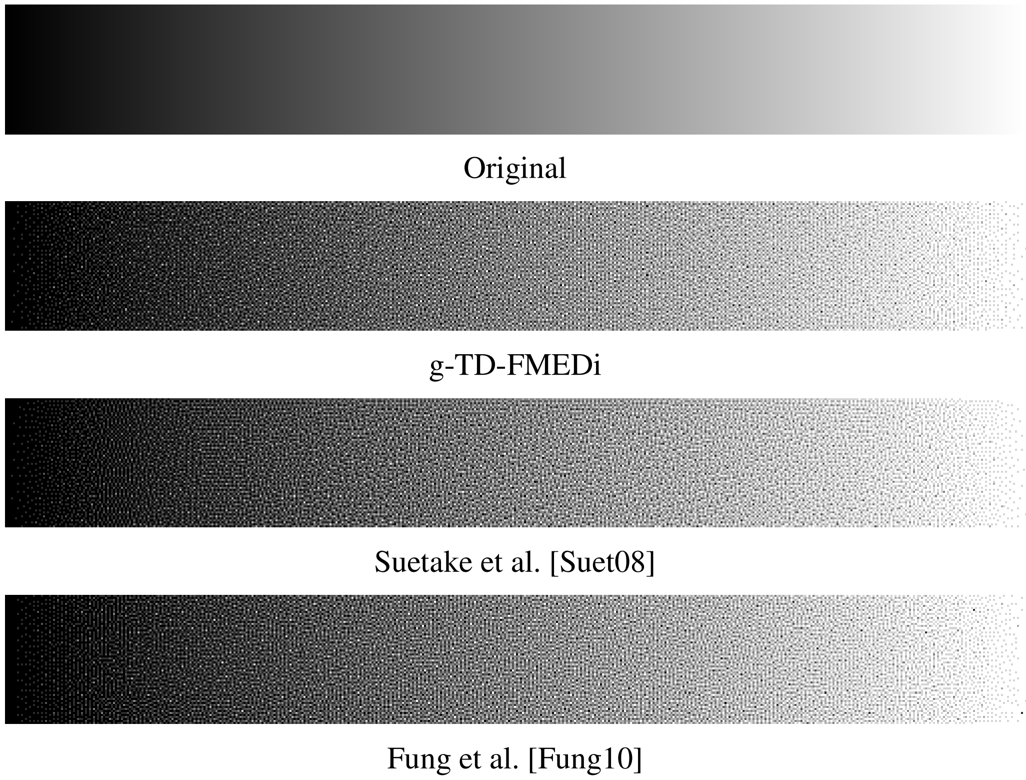


Figure 5.8 Seven-level halftoning outputs of gray-scale ramp obtained with different algorithms

5.4 Summary

In this chapter, we have generalized the idea presented in Chapter 3 and proposed a TD-based multitoning algorithm to produce multitones of any output levels. As human is more sensitive to the dots of higher contrast, at any particular moment during the multilevel halftoning process, we should handle the layers of maximum energy and minimum energy at the same time. In the proposed algorithm, the maximum and the minimum energy layers that have not yet been processed are paired up and handled simultaneously in an interleaving manner. By doing so, the bias introduced by the conventional TD-based algorithms is automatically removed. Besides, the dots of larger contrast can be positioned earlier to improve the visual quality of the output. The suggested change can be easily supported by FMED to provide an output of high quality. Simulation results have showed that the proposed method can provide multitoning outputs of better quality as compared with conventional TD-based multitoning algorithms such as [Suet08], [Rodr08] and [Fung10].

Chapter 6

Conclusions

6.1 Summary of work

Digital halftoning is a technique used to convert a continuous image into a binary image for being displayed or printed. Nowadays, even the printing technology has been developed a lot and a printing system can produce printouts of more than two output levels. Multilevel halftoning is hence required to produce multitones of pleasant quality. Threshold Decomposition is a framework that can be used with a binary halftoning algorithm to produce a multitone of good quality. Under this framework, an input image is decomposed into layers such that they can be processed with a binary halftoning algorithm one by one under a stacking constraint. On one hand this provides flexibility and reduces the realization complexity of multitoning. On the other hand, that it processes layers in turn favors the layers being processed first and hence introduces a bias to position those darker dots in the multitoning output first. This makes it difficult to preserve the bright spatial features of the input image. In this thesis, we have addressed this issue and proposed several remedial solutions to remove the bias. With these solutions, one can produce multitones of even better quality.

As a preliminary study of the addressed issue, in Chapter 3 we have put our focus on how to produce a three-level multitone under the Threshold Decomposition framework and suggested a solution to remove the bias. In three-level TD-based multitoning, an image is decomposed into two layers. Instead of processing them one by one, we have suggested processing them simultaneously in an interleaving manner with FMED. Accordingly, the black dots and white dots are alternately put into the output to generate a three-level multitone. This eliminates the bias and allows the resultant multitone to preserve both the dark and bright spatial image features. A three-level TD-based multitoning algorithm referred to as TD-FMEDi is developed based on the solution proposed in Chapter 3.

In Chapter 4, we continue our preliminary study on the addressed issue based on three-level multitoning. The solution that is proposed in Chapter 3 still cannot remove the bias from the root. When a particular layer is processed, the other layer is not taken into account. The binary halftoning processes for individual layers cannot be jointly optimized simultaneously. To solve this problem, we combined the two layers into a complex energy plane. Without having any predetermined preference, we determine the optimal type and the optimal location of the dot to be put simultaneously based on the complex energy plane at the moment we assign a dot. By doing so, both layers can be taken into account at any time when they are processed and it completely eliminates the bias from the root. A three-level TD-based multitoning algorithm referred to as TD-CMED is developed based on the solution proposed in Chapter 4.

The two remedial solutions proposed in Chapters 3 and 4 are dedicated for three-level multioning. When the number of allowable output levels of a multitone is more than 3, it may not be possible for one to handle all layers at a time. In Chapter 5, we extend the idea of the three-level TD-based multitoning algorithm proposed in Chapter 3 to develop a generalized multitoning algorithm for producing multitones of any number of output levels. In particular, layers are paired up and processed pair by pair. At any particular stage, we select the layer of maximum energy and the layer of minimum energy from the layers that have not yet been processed, and process them in an interleaving manner. This arrangement removes the bias and allows us to position dots of largest contrast first to improve the visual quality of the resultant multitone. It works with FMED [Chan04] to form a multitoning algorithm referred to as g-TD-FMEDi which is able to produce high quality multitones of any number of output levels. Simulation results showed that the proposed method can provide multitoning outputs of better quality as compared with conventional TD-based multitoning algorithm such as [Suet08], [Rodr08] and [Fung10].

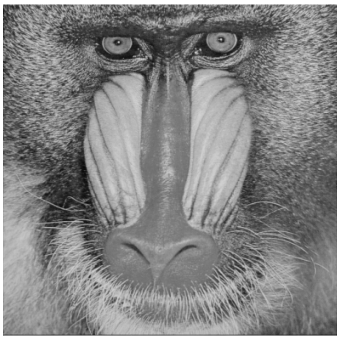
6.2 Future work

Conventional TD-based multitoning algorithms decompose an image into layers and process them sequentially under a stacking constraint. The order that the layers are processed introduces a bias to favor particular types of dots in the resultant multitone, which significantly affects the quality of their outputs. Obviously, whether we can eliminate this bias is critical in improving the quality of the outputs. We have proposed several solutions to remove the bias in this work, but they still have their limitations. In particular, TD-FMEDi and TD-CMED can only produce three-level multitones. G-TD-FMEDi is a generalized version of TD-FMEDi and is able to produce multitones of any number of output levels. However, though it is able to remove the bias to a certain extent, it does not take all layers into account at the time it processes a layer and hence it cannot jointly optimize the positioning of all kinds of ink dots.

In TD-CMED, two layers are combined to form a complex energy plane for being processed such that both layers can be taken care of simultaneously when the complex plane is being processed. This idea may be extended to produce multitones of more than 3 output levels. In particular, it is worthwhile to explore if we can combine all layers into a plane of vectors that contains all information of the layers and search the vector plane to determine the type of the dot that we should assign at a particular moment and the location that we should assign the dot simultaneously. This would be a proper direction to reach the ultimate solution of the problem that we address in this thesis.

Appendix

Testing images of size 512×512 [Spri]



Baboon

Barbara

Boat

Lena

Peppers

Airplane

Goldhill

Girl

Man

References

[Anas88] D. Anastassiou and S. Kollias, “Digital image halftoning using neural networks,” *Proc. SPIE Visual Communications and Image Processing,* Vol. 1001, pp. 1062-1069, Oct 1988.

[Baye73] B. E. Bayer, ‘‘An optimum method for two-level rendition of continuous-tone pictures,’’ *Proc. IEEE Int. Conf. Commun.*, pp. 26-11, 26-15, 1973.

[Baqa98] F. A. Baqai, J. P. Allebach, “Printer models and the direct binary search algorithm,” *Proc. IEEE*, Vol. 5, pp. 2949-2952, 1998.

[Chan98] Y. H. Chan, “A modified multiscale error diffusion technique for digital halftoning,” *IEEE Signal Process. Lett.*, Vol. 5, No. 11, pp.277–280, 1998.

[Chan99] Y. H. Chan, H. T. Kwong, K. T. Lo and C. K. Li, “An error diffusion technique with reduced directional hysteresis,” *Conference of IEEE ISCAS*, pp. 263-266, 1999

[Chan04] Y. H. Chan and S. M. Cheung, “Feature-preserving multiscale error diffusion for digital halftoning,” *J. Electron. Imag.*, Vol. 13, No. 3, pp.639–645, 2004

[Crou93] K. R. Crounse, T. Roska and L. O. Chua, “Image halftoning with cellular neural networks,” *IEEE Trans. Circuits and Systems II: Analog and Digital Signal Processing*, Vol. 40, No.4, pp. 267–283, Apr. 1993.

[Dame01] N. Damera-Venkata and B. L. Evans, “Adaptive threshold modulation for error diffusion halftoning,” *IEEE Trans. Image Process.*, Vol. 10, No.1, pp. 104–116, 2001.

[Floy75] R. W. Floyd and L. Steinberg, “An adaptive algorithm for spatial grey scale,” *in Proc. SID Int. Symp. Digest of Technical Papers.* New York, pp. 36–37, 1975.

[Floy76] R. Floyd and L. Steinberg, “An adaptive algorithm for spatial grayscale,” *Proc. SID*, pp. 75-77,1976.

[Fung10] Y. H. Fung and Y. H. Chan, “Multilevel Halftoning using Multiscale Error Diffusion,” *Journal of Electronic Imaging*, Vol.19, Issue 3, 2010.

[Geis93] R. Geist, R. Reynolds and D. Suggs, “A Markovian framework for digital halftoning,” *ACM Trans. On Graphics*, Vol12, pp.136-159, 1993.

[Gent90] R. S. Gentile, E. Walowit, and J. P. Allebach, “Quantization and multi-level halftoning of color images for near-original image quality,” *J. Opt. Soc. Am. A* 7(6), pp. 1019–1026, 1990.

[Gran90] E. M. Granger, J. F. Hamilton, and L. G. Wash, ‘‘Electronic graphic arts screener,’’ *U.S. Patent No. 4,918,622 assigned to Eastman KodakCompany* ,1990.

[Guo11] J. M. Guo and C. C. Su, “Improved Block Truncation Coding Using Exteme Mean Value Scaling and Block-Based High Speed Direct Binary Search,” *IEEE Signal Process Lett,* Vol. 18, No. 11, pp.694-697,2011.

[Hwan04] B. W. Hwang, T. H. Kang and T. S. Lee, "Distortion-Free of General Information with Edge Enhanced Error Diffusion Halftoing," *Conference of ICCSA, LNCS 3046*, pp. 281-290, 2004

[Jarv76] J. F. Jarvis, C. N. Judice, and W. H. Ninke, "A survey of techniques for the display of continuous-tone pictures on bilevel display," *Comput. Graphics Image Processing,* vol. 5, pp. 13–40, 1976.

[Kats97] I. Katsavounidis and C. C. J. Kuo, "A multiscale error diffusion technique for digital halftoning," *IEEE Trans. Image Process.*, Vol. 6, No.3, pp. 483–490, Mar. 1997.

[Kang99] H. R. Kang, Digital Color Halftoning, SPIE, Bellingham, WA, 1999.

[Klen70] R. J. Klensch, D. Meyerhofer, and J. J. Walsh, ''Electronically generated halftone pictures,'' *RCA Rev.* **31**, pp. 517–533, 1970.

[Lau98] D. L. Lau, G. R. Arce, and N. C. Gallagher, "Green-noise digital halftoning," *Proc. IEEE*, vol. 86, no. 12, pp. 2424–2444, Dec. 1998.

[Lau03] D. L. Lau, R. Ulichney and G. R. Arce, "Blue and green noise halftoning models," *IEEE Signal Processing Magazine.*, Vol. 20, No.4, pp. 28–38, 2003.

[Lau08] D. L. Lau and G. R. Arce*, Modern Digital Halftoning*. New York:Marcel Dekker, 2008.

[Leon92] J. Leone and K. E. Spaulding, ''Method for generating halftone image data with randomly selected threshold array,'' *U.S. Patent No.5,150,428 assigned to Eastman Kodak Company*, 1992.

[Li00] P. S. Li, J. P. Allebach, "Look-up-table based halftoning algorithm," *IEEE Trans. Image Process.*, Vol. 9, No.9, pp. 1593–1603, Sep 2000.

[Lin97] Y. Y. Lin and T. C. Ko, "A modified model-based error diffusion," *IEEE Signal Process. Lett.*, Vol. 4, No. 2, pp.36-38, 1997.

[Lin01] G. Y. Lin and J. P. Allebach, "Multilevel screen design using direct binary search," *Proc. SPIE*, Vol 4663, pp.264-277, Dec 2001.

[Lin02] G. Y. Lin and J. P. Allebach, "Multilevel screen design using direct binary search," *J. Opt. Soc. Am. A* 19(10), pp.1969–1982 ,2002.

[Lui07] K. C. Lui, Y. H. Fung and Y. H. Chan, "A Low-complexity High performance Multiscale Error Diffusion Technique for Digital Halftoning," *Journal of Electronic Imaging*, Vol.16, Issue 1, 013010, 12pages, Jan-Mar 2007.

[Maku95] A. Makur and M. R. Raghuveer, "Digital Image Halftoning by noise thresholding," *Conference of ICASSP*, Vol. 4, pp2535-2538, 1995.

[Mese02] M. Mese and P. P. Vaidyanathan, "Recent advances in digital halftoning and inverse halftoning methods," *IEEE Trans. Circuits and Systems-I*, pp.790-805, Jun. 2002.

[Mits91] T. Mitsa and K. Parker, "Digital halftoning using a blue-noise mask," *in Proc. SPIE, Image Processing Algorithms and Techniques II*, vol. 1452, pp. 47–56,1991.

[Mits92] T. Mitsa and K. J. Parker, "Digital halftoning technique using a blue noise mask," *J. Opt. Soc. A*, vol. 9, no. 11, pp. 1920–1929, 1992.

[Ostr00] V. Ostromoukhov, “Optimal halftoning for network-based imaging,” *Proc. SPIE, Color Imageing: Device-Independent Color, Color Hardcopy, and Graphic Arts VI, Reiner Eschbach, Vol. 4300, pp.438-443*, Dec 2000.

[Pang08] W. M. Pang, Y. QU, T. T. Wong, D. Cohen-Or and P. A. Heng, “Structure-aware halftoning,” *Conference of SIGGRAPH*, vol89, pp.1-8, 2008

[Papp03] T. N. Pappas, J. P. Allebach and D. L. Neuhoff, “Model-based digital halftoning,” *IEEE Signal Process. Magazinet.*, Vol. 20, No. 4, pp.14-27,2003.

[Park07] T. Y. Park et al., “Banding-artifact reduction using an improved threshold scaling function in multitoning with stochastic screen,” *J. Imag. Sci.Technol*. 51(6), pp.502–508, 2007.

[Rodr08] J. B. Rodriguez, G. R. Arce and D. L. Lau, “Blue-Noise Multitone Dithering,” *IEEE Trans. Image Process.*, Vol. 17, No.8, pp. 1368–1382, Aug. 2008.

[Sara12] G. Sarailidis and I. Katsavounidis, “A multiscale error diffusion technique for digital multitoning,” *IEEE Trans. Image Process*. 21(5), pp.2693–2705 ,2012.

[Schr86] W. F. Schreiber, *Fundamentals of Electronic Imaging Systems: Some Aspects of Image Processing*. New York: Springer-Verlag, 1986.

[Spri] The Springer Images, http://www.springerimages.com/

[Stev85] R. L. Stevenson and G. R. Arce, “Binary display of hexagonally sampled continuous-tone images,” *J. Opt. Soc. Amer. A.,* vol. 2, pp. 1009–1013, 1985.

[Stuc81] P. Stucki, “Mecca—A multiple-error correcting computation algorithm for bilevel image hardcopy reproduction,” *IBM Res. Lab., Res. Rep*. RZ1060, 1981.

[Suet08] N. Suetake, M. Sakano, E. Nakashima and E. Uchino, “Simple multilevel halftoning excelled in gradation reproducibility,” *Optics Letters*, Vol. 33, No. 4,pp.339-341, 2008.

[Ulic87] R. A. Ulichney, *Digital Halftoning. Cambridge, MA: MIT Press*, 1987.

[Ulic93] R. A. Ulichney, “The void-and-cluster method for dither array generation,” *Proc. SPIE, Human Vision, VisualProcessing and Digital Display IV,* vol. 1913, pp. 332–343, 1993.

[Wang02] Z. Wang and A. C. Bovik, “A universal image quality index,” *IEEE Signal Process. Lett.*, vol. 9, no. 3, pp. 81–84, Mar. 2002.

[Wang04] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *Image Processing, IEEE Trans. Image Process*. Vol.13, No.4, pp.600-612, 2004.

[Yu99] Q. Yu et al., “Digital multitoning with overmodulation for smooth texture transition,” *J. Electron. Imag*. 8(3), pp.311–321 ,1999.

[Zeng00] H. Zeng, H. Ying and X. B. Sun, “A new approach of halftoning based on error diffusion with rough set filtering,” *Conference of WCICA*, Vol.4, pp.2544-2548, 2000.

[Zhen10] H. Zhen, “Hierarchical Colorant-Based Direct Binary Search Halftoning,” *IEEE Trans. Image Process*. 19(7), pp.1824-1836 ,2010.

[Zhou11] Y. Zhou, C. Chen, Q. Wang and J. Bu, “Block-based threshold modulation error diffusion,” *J. Electron. Imag.*, Vol. 20, No. 1, 2011